Programma verzameling voor de MSX-computers

MSXB1

DATA BECKER NEDERLANDS*

Lüers

Programma-verzameling voor de MSX-computers

Met complete utilities

MSX Bibliotheek 1

DATA BECKER NEDERLANDS*

Lüers

Programma-verzameling voor de MSX-computers

Met complete utilities

MSX Bibliotheek 1

DATA BECKER
NEDERLANDS*

1e druk 1985
2e druk 1986

Oorspronkelijke titel: MSX Programmsammlung
Copyright © 1984 Data Becker GmbH, Düsseldorf
Voor de Nederlandse vertaling:
Copyright © 1985 Uitg. A. W. Bruna & Zoon, Utrecht
Omslagontwerp: Martin van Keulen
Vertaling: ED DATA DESIGN, Hoog-Karspel
Produktie: A. W. Bruna en Zoon, Utrecht
Redaktie: A. W. Bruna en Zoon, Utrecht
ISBN 90 229 3350 4
D/1986/0939/87

De keuze en opbouw van de programma's die in dit boek voorkomen, zijn gebaseerd op hun educatieve waarde. Alle programma's zijn getest op hun juiste werking. De uitgever kan geen enkele verantwoordelijkheid voor eventueel toch optredende fouten aanvaarden.

Uit deze uitgave mag niets worden verveelvoudigd en/of openbaar gemaakt door middel van druk, fotokopie, microfilm of op welke andere wijze ook zonder voorafgaande schriftelijke toestemming van de uitgever.

Boeken en programma's van

DATA BECKER
NEDERLANDS*

worden in de handel gebracht door:
A. W. Bruna & Zoons Uitgeversmij b.v.
Postbus 8411, 3503 RK Utrecht
en
A. W. Bruna & Zoon n.v.
Antwerpsesteenweg 29a, 2630 Aartselaar.

Beste MSX-vrienden en MSX-belangstellenden.

Tot op heden keek elke particulier, die na rijp beraad tot de aankoop van een homecomputer had besloten, afgunstig naar de grotere computers. Want in deze categorie heeft reeds jaren geleden standaardisering plaatsgevonden. Namen als 'CP/M' en 'MS-DOS', evenals het begrip IBM-compatibel, stonden daar voor in en tegenwoordig geldt het in steeds sterkere mate, dat de programmatuur van computer X ook op computer Y zonder problemen draait. En omgekeerd geldt uiteraard hetzelfde. Daar bestaan geen problemen in de trant van: 'Waarom kan ik deze databank uitgerekend niet op mijn computer laten werken?!'.

Hoe staan de zaken er tegenwoordig bij de meeste homecomputers voor? Fabrikant X brengt een computer op de markt waarop natuurlijk alleen de randapparatuur en de programmatuur van deze ene fabrikant past. Gevolg: de aankoop van computer X resulteert normaal gesproken ook in een aanzienlijk aantal vervolgaankopen bij dezelfde fabrikant. Hieraan hebben velen reeds goudgeld verdiend. Is de markt met produkt X halverwege het verzadigingspunt, dan brengt dezelfde fabrikant een produkt Y uit, waardoor opnieuw de kassa begint te rinkelen door de vervolgaankopen. Er worden nu eenmaal nieuwe aansluitingen aan de computer aangebracht, die ook weer niet genormaliseerd zijn. En waar draait het voor de arme fabrikant op uit? De software van produkt X loopt begrijpelijkerwijze ook niet meer op computer Y. Of deze fabrikant ooit eens op zijn gezicht zal vallen met deze politiek? We zullen het moeten afwachten. De MSX-computers kunnen, sterker nog, moeten, als overwinnaar over de eindstreep gaan. Want hier wordt niemand meer voor de gek gehouden.

Hoe is zoiets tot stand gekomen? Alle Japanse bedrijven uit de hifi- en videobranche zijn enige tijd geleden gezamenlijk om de tafel gaan zitten en hebben overwogen hoe de economische grootmachten als de Verenigde Staten en Europa weer eens gigantisch afgetroefd zouden kunnen worden. (Zie ook de ontwikkeling bij de camara's, de hifi- en videosector en niet op de laatste plaats de automobielindustrie).

Omdat men de homecomputermarkt niet als eerste wilde betreden, moest nu gezamenlijk iets bijzonders uitgebroed worden. Ten slotte stelde men zich ten doel homecomputers op de markt te brengen die niet alleen compatibiliteit (uitwisselbaarheid) bij de software (programmatuur) maar ook bij de hardware (aansluiten van randapparatuur) moesten garanderen.

Men bracht dit allemaal samen onder de noemer 'MSX'. Een afkorting voor de

krachtigste BASIC, die weer van 'MICRO-SOFT', een zeer gerenommeerd Amerikaans software-huis, afkomstig is: 'Microsoft Super eXtended basic'.

Maar de MSX-standaard omvat meer: niet alleen de Basic is hetzelfde, ook de chips voor beeld en geluid (over dit centrale thema bestaat overigens een uitgebreid DATA BECKER-boek: MSX Sound und Grafik) lijken bij alle MSX-computers als twee druppels water op elkaar. Bovendien heeft elke MSX-computer, zoals het zich nu laat aanzien, een gestandaardiseerde Centronics-poort. Het gebruik van commerciële CP/M-programma's (eigenlijk is deze software van de grotere computers afkomstig) behoort voor 100% tot de mogelijkheden. En lest best. Het MSX diskette operatingsystem MSX-DOS is datacompatibel met MS-DOS van IBM. Hetgeen inhoudt dat u uw gegevens van kantoor mee naar huis kunt nemen en in alle rust op uw MSX-computer met aangesloten diskdrive verder kunt werken.

U heeft misschien al wel in de gaten dat dit uitgangspunt van de Japanners iets is waar u staat op kunt maken. Dit idee moet eenvoudigweg succes boeken. Dat kan gewoon niet anders.

Nog twee dingen als aanvulling:

1) Omdat de fabrikanten van MSX-computers bijna allemaal uit de hifi- en videobranche afkomstig zijn, is het aansluiten van de MSX-computers aan bij voorbeeld de videorecorder, de beeldplatenspeler, de bandrecorder en dergelijke niet ver meer (voorbeelden hiervan waren er voor iedereen al te zien op de Hifivideo te Düsseldorf in 1984). Resumerend: een homecomputersysteem dat werkelijk in het dagelijkse leven thuishoort en zichzelf het snelst aan zijn omgeving aanpast.

2) Niet alleen de gebruiksmogelijkheden van de MSX-computer zijn aanlokkelijk! Ook de technische prestaties zoals de geheugenuitbreiding tot 256 KB ram, 32 sprites, 16 kleuren, grootoplossend kleurenbeeld met 256*192 beeldpunten en een ingebouwde driestemmige soundprocessor spreken boekdelen voor de computerkenner!

Laten we maar eens afwachten hoe de markt op zijn kop komt te staan. De Japanners zijn vertegenwoordigd door meer dan twintig firma's met gerenommeerde namen zoals Sony, Spectravideo, Hitachi, JVC enz. De Koreanen o.a Goldstar en Samsung en ook Philips in Nederland verwachten dat alleen al in 1985 de helft van alle verkochte homecomputers het MSX-embleem zullen dragen. Toi, toi, toi MSX!

P.S.: De begrippen 'homecomputer' en 'personalcomputer' zijn inmiddels zo nietszeggend, dat we bij alle MSX-computers in ieder geval ook van personalcomputers kunnen spreken.

```
MSX BASIC version 1.0
Copyright 1983 by Microsoft
28815 Bytes free
Ok
*






color auto goto list run
```

Beste lezer

U heeft het boek 'MSX verzamelde programma's' in handen en wilt zich nu misschien snel een oordeel over dit boek vormen. Daarom heeft u zeker al een blik op de inhoudsopgave geworpen... goed zo... en nu wilt u nog even snel het voorwoord doorvliegen... en dan in een rustig hoekje beginnen met het programmeren ofwel de programma's in uw MSX-computer intoetsen!

Geef me echter de kans nog een paar dingen aan te stippen, voordat u echt van start gaat.

Achter deze verzamelde programma's steekt een ander concept dan u in eerste instantie misschien zult vermoeden. Deze programmaverzameling bestaat namelijk niet alleen uit programma's, maar ook uit zo'n 53 pagina's tekst. Niet dat ik zonder programma's kwam te zitten of dat mij geen idee meer te binnen schoot. Zou ik meteen alles gepubliceerd hebben, dan had dit boek minstens 400 pagina's geteld. Nee, u moet volgens mij niet alleen maar overtypen, overtypen en nog eens overtypen... Nee, u moet ook nog wat over uw MSX-computer leren.

Weliswaar wordt niet elk programma regel voor regel verklaard. Daarvoor ontbreekt de ruimte en misschien ook wel uw uithoudingsvermogen. Maar op interessante gedeelten wordt in de voorafgaande tekst nader ingegaan en voor het merendeel zijn de programma's, als toelichting, ook nog van zgn. REM-statements, ofwel REM-commando's voorzien.

Ook inhoudelijk ligt het zwaartepunt meer op de praktische bruikbaarheid, zodat u muzikaal of grafisch iets kunt ontwerpen, wat aan planning kunt doen of uw grammofoonplaten of de voetbaluitslagen met uw MSX-computer kunt bijhouden.

Tenslotte ligt er nog een zwaartepunt in deze verzamelde programma's. U moet een beetje een indruk krijgen hoe uw MSX-computer 'intern' functioneert. De programma's zoals 'disassembler' en 'variabelenreferentielijst' zijn daar slechts twee voorbeelden van.

In elk geval, en laat ik daar mee besluiten, moet u er plezier aan beleven om in de geheimzinnige wereld van de computer door te dringen. En speciaal in die van uw MSX-computer. En wie weet groeit u door de 'MSX verzamelde programma's' wel uit tot een programmeur.

Tot slot: Veel plezier met het werken toegewenst door de auteur van de programma's en de tekst van dit DATA BECKER boek.

MSX en Spectravideo 318/328

Omdat de computers SVI-318 en SVI-328 reeds op de markt verschenen voordat definitieve overeenstemming was bereikt over wat MSX eigenlijk zou moeten inhouden, moeten de modellen SVI-318 en SVI-328 als voorlopers van de MSX-computer gezien worden. Er zijn twee mogelijkheden om de programma's uit dit boek ook op de SVI-318 (voor zover de geheugencapaciteit dit toelaat) en de SVI-328 te laten werken:

1) De firma Spectravideo levert een zgn. module voor de 318 en de 328, die van beide apparaten voor de volle 100% een MSX-computer maakt.

2) Aansluitend aan deze toelichting heb ik in 7 punten samengevat wat u in de programma's van deze MSX verzamelde programma's moet veranderen, zodat ook u als SVI-318/328 gebruiker deze programma's kunt benutten.

Bovendien brengt de firma Spectravideo nog de SVI-728 op de markt, die wel een volledige MSX-computer is.

Verschillen tussen MSX- en Spectravideo 318/328 BASIC

1) In- en uitschakelen van toetsenbordklik en functietoetsenindicatie

MSX

Klik: 3. Parameter bij SCREEN-commando op "1" (aan) of "0" (uit).
Functie-indicatie: "KEY ON" (aan) en "KEY OFF" (uit).

SVI-318/328

Klik: Commando "CLICK ON" (aan) of "CLICK OFF" (uit).
Functietoetsenindicatie: bij "SCREEN" 2. Parameter op "1" (aan) of "0" (uit) - alleen bij "SCREEN 1".
"KEY ON" en "KEY OFF" betekent: interuptrequest van de functietoetsen aan of uit (bij MSX niet aan- of uitschakelbaar)

2) Het 'SCREEN'-commando

MSX

SCREEN 0: 40 tekens per regel. Door "WIDTH" tot 1 teken per regel.
SCREEN 1: 32 tekens per regel. Door "WIDTH" tot 1 teken per regel.
SCREEN 2: hoog oplossend vermogen 256*192 beeldpunten.
SCREEN 3: laag oplossend vermogen 64*48 beeldpunten.

SVI-318/328

SCREEN 0: 40 tekens per regel. Door "WIDTH" tot 39 tekens per regel.
SCREEN 1: hoog oplossend vermogen 256*192 beeldpunten.
SCREEN 2: laag oplossend vermogen 64*48 beeldpunten.

3) Het afbeelden van tekst in de grafische mode.

MSX

'SCREEN 2:'
'OPEN "GRP:" FOR OUTPUT A$ #1:'
Bijvoorbeeld 'DRAW "BM 128,96":' of 'PRESET (128,96):'
'PRINT #1, "MSX"'
Per regel zijn totaal 32 opeenvolgende tekens samen te stellen.

SVI-318/328

'SCREEN 1:'
'LOCATE 128,96:'
'PRINT "SVI-318/328'
Per regel zijn totaal 42 opeenvolgende tekens samen te stellen.

4) Tekst in reverse ofwel negatief

MSX

Dit kan alleen met een hulpprogramma bereikt worden.

SVI-318/328

Reverse inschakelen: 'PRINT CHR$(27);"p";'
Reverse uitschakelen: 'PRINT CHR$(27);"q";'

5) Accenttekens in de karakterset.

MSX

In de voor de Europese markt bestemde MSX-computers zijn diverse leestekens reeds in de karakterset aanwezig zij het op de verkeerde plaats wat betreft het gebruik met een andere dan een standaard MSX-printer.

'[' in het video-RAM-geheugen bevindt zich op geheugenlocatie 2776.

De karakterset is zonder verstoring van de ASCII-tekens te veranderen, vanaf geheugenplaats 3072 in het video-RAM-geheugen.

SVI-318/328

In de karakterset zijn geen leestekens voorhanden. Ze moeten zelf worden samengesteld.

'[' in het video-RAM-geheugen bevindt zich op geheugenlocatie 2520.

De karakterset is zonder verstoring van de ASCII-tekens te veranderen, vanaf geheugenplaats 3584 in het video-RAM-geheugen.

6) Besturingskarakters

Behalve de onderstaande zijn de besturingskarakters van de MSX en SVI-318/328 gelijk.

MSX

199	SPRITE	200	VDP	201	BASE	202	CALL
203	TIME	204	KEY	205	MAX	206	MOTOR
207	BLOAD	208	BSAVE	238	>	239	=
240	<	241	+	242	-	243	*
244	/	245	^	246	AND	247	OR
248	XOR	249	EQV	250	IMP	251	MOD

252 tot en met 254: geen zichtbare tekens.

SVI-318/328

199	KEY	200	CLICK	201	SWITCH	202	MAX
203	MON	204	MOTOR	205	BLOAD	206	BSAVE
207	MDM	208	DIAL	238	SPRITE	239	TIME
240	>	241	=	242	<	243	+
244	-	245	*	246	/	247	^
248	AND	249	OR	250	XOR	251	EQV
252	IMP	253	MOD	254			

7) Error meldingen.

Behalve de onderstaande zijn de error-meldingen van de MSX en SVI-318/328 gelijk.

MSX

ERROR 58: 'Sequential I/O only'
ERROR 59: 'File not open'
ERROR 60: 'Unprintable error'

SVI-318/328

ERROR 58: 'Sequential after put'
ERROR 59: 'Sequential I/O only'
ERROR 60: 'File not open'

Geheugeninhoud 1

Bekijken we het resultaat van dit programma op het beeldscherm, dan constateren we dat de geheugenplaats waarvan we de inhoud met het commando 'PEEK' opvragen niet overal leeg ofwel '0' is.

Wel verschijnen op het beeldscherm naar het zich laat aanzien allemaal willekeurige getallen. Maar we zullen meteen vaststellen, dat achter deze getallen voor een deel zinvolle letters en tekens schuilgaan. Ter verduidelijking een experiment:

Toets het volgende in uw MSX-computer in:

PRINT ASC("!")' ENTER

Resultaat: 33. Dit houdt in, dat zich achter het uitroepteken voor de computer het getal 33 schuilhoudt. Laten we het eens andersom proberen:

Toetst u maar eens in:

PRINT CHR$(33)' ENTER

Resultaat: '!'. Hiermee hebben we de computer opdracht gegeven het getal 33 weer om te zetten in een karakter. Het resultaat van het getal 33 is ons uitroepteken!

Op deze manier kunnen we ook onze getallenreeks uit het programma geheugeninhoud 1 leren begrijpen. Toets ter controle meteen maar eens het programma geheugeninhoud 2 in.

Gebruiksaanwijzing voor het programma 'geheugeninhoud 1'

Start- en eindadres moeten niet verder dan 140 tekens van elkaar verwijderd zijn. Alleen dan verkrijgen we alle gewenste data op een beeldscherm.

```
Beginadres ingeven ? 1000
Eindadres ingeven ? 1140

255  255  47  87  121  211  168  34
197  252  235  34  199  252  237  86
195  128  38  58  177  251  167  192
229  33  155  252  243  126  54  0  225
 251  167  200  254  3  40^ 28  229
213  197  205  218  9  33  155  252
243  126  54  0  251  167  40  248  245
 205  39  10  241  193  209  225  254
3  192  229  205  104  4  205  84  4
488 10  33  106  252  243  205  241  14
 251  225  201  205  59  8  58  193
252  38  64  205  94  2  225  175  237
123  177  246  197  195  230  99  58
106  252  15  208  42  107  252  124
181  200  42  28  244  35  124  181
200  55  201  42  248  243  34  250
243  201  219  170  230  240  246  7

Wilt u meerdere delen van het
geheugen onderzoeken ( /N) ? *
```

```
10 REM Geheugeninhoud 1
20 REM MSX VERZAMELDE PROGRAMMA'S
30 REM Copyright 1984 DATA BECKER &
                             Rainer Lueers
40 COLOR 15,1,15:SCREEN 0:KEY OFF
50 WIDTH 40
60 REM Boven- en onderwaarde begrenzen
70 INPUT "Geef het startadres op";A
80 INPUT "Geef het  einadres  op";E
90 IF E<A OR E>65535! OR A<0 OR A>65535!
 THEN GOSUB 210: GOTO50
100 PRINT
110 REM Weergave van de begrensde geheug
eninhoud d.m.v. getallen.
120 FOR N=A TO E
130 PRINT PEEK(N);
140 NEXT N
150 PRINT
160 PRINT
170 INPUT"Wilt u het geheugen nog verder
           onderzoeken ( /N) ";F$
180 IF LEFT$(F$,1)="n" OR LEFT$(F$,1)="N
" THEN F$="N"
190 IF F$<>"N" THEN GOTO 10
200 END
210 PRINT:PRINT TAB(10) CHR$(200) " VERK
EERDE OPGAVE! " CHR$(200)
220 GOSUB 230:RETURN
230 PRINT:PRINT TAB(16) "<Druk een toets
 a.u.b.>"
240 F$=INKEY$:IF F$="" THEN GOTO 240
250 RETURN
```

Geheugeninhoud 2

Nadat u dit programma heeft ingetypt en met 'RUN' gestart, gebeurt het volgende: wat u ook doet, er gebeurt helemaal niets of er gebeurt iets onverwachts (het beeldscherm wordt gewist, uw MSX-computer begint wild te 'beepen'). Hoe kan dat nou?

Zoals reeds in het vorige programma werd aangetoond, bestaan er zinvolle direct af te beelden tekens. CHR$(32) is een spatie t/m CHR$(126), dat is bijvoorbeeld een slingerlijntje. Van CHR$(127) tot CHR$(255) bevinden zich de grafische tekens. Er zijn echter ook andere CHR$-tekens die niet zo gemakkelijk af te beelden zijn. CHR$(12) is het beeldscherm wissen. CHR$(7) is de 'BEEP' laten horen. Zie voor een verdere specificatie het handboek van uw MSX-computer.

Daarom moeten we nog een derde programma schrijven om alleen die getallen als tekens op het beeldscherm te laten afbeelden, die op deze manier ook daadwerkelijk af te beelden zijn. Dat gaan we nu doen met een programma-uitbreiding in het programma 'geheugeninhoud 3'.

Gebruiksaanwijzing voor het programma 'geheugeninhoud 2'

Wanneer op het beeldscherm niets behoorlijks meer te herkennen valt en u het programma nog verder wilt gebruiken, start dan het programma 'geheugeninhoud 2' gewoon opnieuw met 'RUN'.

```
10 REM Geheugeninhoud 2
20 REM MSX VERZAMELDE PROGRAMMA'S
30 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
40 COLOR 15,1,15:SCREEN 0:KEY OFF
50 WIDTH 40
60 REM Boven- en onderwaarde begrenzen
70 INPUT "Geef het startadres op";A
80 INPUT "Geef het einadres  op";E
90 IF E<A OR E>65535! OR A<0 OR A>65535!
 THEN GOSUB 210: GOTO50
100 PRINT
110 REM Weergave van de begrensde geheug
eninhoud d.m.v. getallen.
120 FOR N=A TO E
130 PRINT CHR$(PEEK(N));
140 NEXT N
150 PRINT
160 PRINT
170 INPUT"Wilt u het geheugen nog verder
           onderzoeken ( /N) ";F$
180 IF LEFT$(F$,1)="n" OR LEFT$(F$,1)="N
" THEN F$="N"
190 IF F$<>"N" THEN GOTO 10
200 END
210 PRINT:PRINT TAB(10) CHR$(200) " VERK
EERDE OPGAVE! " CHR$(200)
220 GOSUB 230:RETURN
230 PRINT:PRINT TAB(16) "<Druk een toets
 a.u.b.>"
240 F$=INKEY$:IF F$="" THEN GOTO 240
250 RETURN
```

Geheugeninhoud 3

Door het invoeren van dit programma worden meteen twee vliegen in één klap geslagen. Enerzijds verschijnt de weergave van de geheugeninhoud nu geordend op het beeldscherm. Achtereenvolgens: startadres, inhoud van dit startadres en de volgende zeven bytes, alsmede de CHR$-betekenis van deze acht bytes. Allemaal op één beeldschermregel. Bovendien worden nu alleen de zinvolle 'CHR$' afgebeeld. Vanaf 32 de spatie t/m 126 de slingerlijn. Voor de niet zonder meer af te beelden 'CHR$' hebben we eenvoudigweg gebruik gemaakt van CHR$(46) ofwel een punt.

Aan u de keuze: of u geeft het startadres op en drukt vervolgens slechts op de (ENTER)-toets. Zodoende worden alleen het startadres en de inhoud van de 8 bytes met de bijbehorende CHR$'s afgebeeld. Wilt u meer dan één regel gegevens afbeelden, dan dient u na het voor de eerste maal indrukken van de (ENTER)-toets dezelfde of een andere toets voortdurend in te drukken. Dan verschijnen telkens acht bytes per regel op het beeldscherm van de hierop volgende geheugenlocatie.

Laten we nu eens kijken waar ons BASIC-programma 'geheugeninhoud 3' gebleven is. Bij de MSX-computers evenals bij elke andere computer worden de commando's voor een deel slechts als een enkel getal ofwel besturingskarakter in het geheugen vastgelegd. Dit in tegenstelling tot de woorden die achter een REM-commando staan. Deze blijven volledig intact. Geeft u als voorbeeld maar eens '32906' als startadres op. Rechts bij de CHR$ weergave moet u nu het woordje 'MSX' kunnen ontdekken. Bekijken we tevens de programmalisting dan ontdekken we het in regel 30.

```
30 REM MSX verzamelde programma's.
```

Dit werkt overigens alleen maar, als u het programma regel voor regel precies zo heeft ingetypt als het in het boek staat. Zonder ook maar één spatie te hebben vergeten!

Als we '32906' als startadres opgegeven hebben en vervolgens op een willekeurige toets drukken, dan zullen we nog andere bestanddelen uit ons programma 'geheugeninhoud 3' tegenkomen.

Gebruiksaanwijzing voor het programma 'geheugeninhoud 3'

Zoals u zeker al opgemerkt zult hebben, wordt hier geen gebruik gemaakt van het ons vertrouwde decimale stelsel (waarbij getallen zijn samengesteld uit de

cijfers 0 t/m 9), maar van het hexadecimale stelsel (samengesteld uit cijfers van 0 t/m F). Dit heeft de volgende reden: enerzijds is het hexadecimale stelsel 'het' stelsel voor de hackers en machinetaalfreaks. Op deze manier willen wij als 'geheugeninhoud-onderzoeker', een ander woord voor hacker, het ook gebruiken! Van de andere kant laat zich in 40 posities met het decimale stelsel niet zoveel informatie onderbrengen als met behulp van het hexadecimale stelsel. ('255' dec. = 'FF' hex. of '65535' dec. = 'FFFF' hex.).

Decimaal omrekenen naar hexadecimaal is met uw MSX-computer heel eenvoudig uit te voeren met het commando 'HEX$('.

Bijvoorbeeld: 'PRINT HEX$(65535)', uitkomst = 'FFFF'.

```
10 REM Geheugeninhoud 3
20 REM Onderzoeken van de geheugeninhoud
       met PEEK en CHR$( in overzicht-
       lijke vorm
30 REM MSX Verzamelde programma's
40 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
50 SCREEN 0:COLOR 15,1,15:KEY OFF
60 WIDTH 40
70 REM Boven- en onderwaarde begrenzen
80 INPUT "startadres opgeven ";A
90 IF A<0 OR A>65535! THEN GOSUB 320:GOT
O 60
100 A$=STRING$(4-LEN(HEX$(A)),"0")+HEX$(
A)
110 REM De volgende regels zorgen voor
        een optimale beeldweergave
        met 40 tekens per regel
120 PRINT A$;" ";
130 REM De beeldschermregel wordt
        telkens met acht bytes gevuld
140 FOR Z=0 TO 7
150 ZZ=PEEK(A+Z):ZZ$=STRING$(2-LEN(HEX$(
ZZ)),"0")+HEX$(ZZ)
160 PRINT ZZ$;" ";
170 NEXT Z
180 REM Afdukken van de afwijkende
        karakterstrings als de PEEK-
        waarde <32 of >126 is
190 FOR Z=0 TO 7
200 ZZ=A+Z
210 ZZ=PEEK(ZZ)
220 IF ZZ<32 THEN ZZ=46
230 IF ZZ>126 THEN ZZ=46
240 PRINT CHR$(ZZ);
250 NEXT Z
260 PRINT
270 A=A+8
```

```
280 A$=INKEY$
290 IF A$="" THEN GOTO 280
300 GOTO 100
310 END
320 PRINT:PRINT TAB(10) CHR$(200) " VERK
EERDE OPGAVE! " CHR$(200)
330 GOSUB 340: RETURN
340 PRINT:PRINT TAB(6) "<Druk een toets
a.u.b.>"
350 F$=INKEY$: IF F$="" THEN GOTO 350
360 RETURN
```

Geheugeninhoud 4

Tot nu toe hebben we het geheugen doorzocht en daarbij voor een deel alleen cijfers 'geheugeninhoud 1', rare tekens 'geheugeninhoud 2' en ten slotte beide in geordende vorm op het beeldscherm gerealiseerd, 'geheugeninhoud 3'. Tijdens het werken met 'geheugeninhoud 3' is ons door het vergelijken van het linker- en rechtergedeelte van het beeld duidelijk geworden dat ons BASIC-programma warempel alleen maar uit getallen bestaat. Getallen die naderhand door uw MSX-computer weer in letters en tekens omgezet worden.

Laten we eens kijken hoe de computer data en programma's in het geheugen vastlegt. Om geheugenruimte te sparen worden BASIC-commando's na het invoeren met ENTER in afkortingen veranderd. Dit gebeurt met de zogenaamde besturingskarakters (zie tevens de appendix). Op deze manier gebruikt het commando 'PRINT' niet vijf bytes, voor vijf letters, maar slechts een enkele byte.

Zetten we echter in een REM-regel 'PRINT' of voeren we in een programmaregel ' PRINT "PRINT" ' in, dan wordt in dat geval de, niet als commando te interpreteren, 'PRINT' letter voor letter in het geheugen opgeslagen. Met andere woorden: hiervoor worden, zoals te verwachten, vijf geheugenplaatsen = vijf bytes in onze MSX-computer gebruikt.

Zoeken we naar het voorkomen van een willekeurig woord in een BASIC-programma, dan kan dat zeer lang duren. Met 65535 geheugenplaatsen is dit vergelijkbaar met het zoeken naar een speld in een hooiberg. Daarom zullen we het wat systematischer moeten aanpakken.

Het programma 'geheugeninhoud 4' helpt ons daarbij aanzienlijk. U geeft een woord bestaande uit niet meer dan zes letters op en geeft de computer de instructie waar hij naar dit woord in het geheugen moet zoeken. Moet hij het hele geheugen doorzoeken, dan duurt het zo ongeveer een eeuwigheid. Moet hij in een door u beperkte geheugenruimte zoeken, dan duurt het korter. Als resultaat wordt in beide gevallen, vooropgesteld dat het trefwoord ook te vinden is, de geheugenlocatie weergegeven. Voor de grap kunt u dan door een 'POKE' (kleiner dan 127 en groter dan 31) iets rechtstreeks veranderen in de aangegeven geheugenlocatie. Een voorbeeld:

In regel 80 staat het woordje 'lengte'.

Allereerst zoeken we de betreffende geheugenlocatie op. Hoe? Het programma starten met 'RUN', de op te zoeken gegevens invoeren = 'lengte' (ENTER).

In het hele geheugen laten zoeken? Deze laatste vraag met 'N' beantwoorden (ENTER). We moeten lang wachten, maar ten slotte verschijnt op het beeldscherm: 'lengte op geheugenlocatie 33054 gevonden'. Komt er bij u een ander getal uit dan is dat niet erg. We noemen het gevonden getal 'GETAL' , in ons geval 'GETAL' = 33054.

Laten we nu eens kijken wat er bij 'GETAL' in het geheugen staat. 'PRINT PEEK(GETAL)'. Alstublieft niet 'GETAL' intypen, maar het getal hiervoor in de plaats gebruiken dat uw MSX-computer u zojuist opgegeven heeft. Bij mij dus: 'PRINT PEEK(33054)'. Uitkomst? '76'! En wat betekent in ons taalgebruik het getal 76? 'PRINT CHR$(76)' = 'L'!

Juist, het getal 76 staat voor de eerste letter van het eerder door ons opgezochte woord 'lengte'. Als we regel 80 listen dan staat daar 'lengte'. Met 'POKE GETAL', een ander getal dan 76!, zullen we nu de 'L' van lengte in ons BASIC-programma in bijvoorbeeld een 'G' veranderen. Alleen, welk getal staat volgens de computer voor de letter 'G'? Wel, dat is er heel gemakkelijk uit te krijgen met het commando 'ASC('. Dus: 'PRINT ASC("G")' = '71'. Nu eenvoudigweg invoeren 'POKE GETAL,71' en tot slot nog eenmaal regel 80 listen. 'Lengte' werd veranderd in 'Gengte'!

Op deze manier leert u een beetje hoe de BASIC-programma's in het geheugen worden vastgelegd en misschien dat u daardoor de complexiteit van een computer zoals uw MSX-computer leert inzien. BASIC is feitelijk het eindprodukt van talrijke interne verbindingen.

Er wordt echter nog een andere dimensie met het programma 'geheugeninhoud 4' ontsluierd. Namelijk het vastleggen van de regelnummers. Start hiertoe nog eenmaal het programma met 'RUN'. Geef opnieuw het trefwoord 'lengte' (of nu 'gengte') op. En laat in 'P=programma' hiernaar zoeken. Na enige tijd staat niet alleen de bijbehorende geheugenplaats weer op het beeldscherm, maar ook het bijbehorende regelnummer van ons BASIC-programma.

Het berekenen hiervan is betrekkelijk eenvoudig, als men maar weet hoe. Ieder MSX-BASIC-programma begint normaal gesproken op geheugenlocatie 32769. Op de geheugenplaatsen 32769 en 32770 staat de geheugenplaats van de volgende regel van ons programma aangegeven. Deze gegevens kunnen we opvragen met: 'PRINT PEEK(32769) = 36' en 'PRINT PEEK(32770) = 128'. Het berekenen gaat als volgt: 'PRINT PEEK(32769) + (PEEK(32770)*256) = 32804'. Hieruit blijkt dat de tweede programmaregel begint op 32804. Hieruit kan men ook de lengte van de huidige regel opmaken, indien men 32769 van 32804 aftrekt. Op geheugenplaats 32771 en 32772 is het nummer van de

eerste programmaregel vastgelegd. Opvragen met: 'PRINT PEEK(32771) = 10' en 'PRINT PEEK(32772) = 0'. Berekenen: 'PRINT PEEK(32771) + (PEEK(32772)*256) = 10'. Dus de eerste regel van ons programma 'geheugeninhoud 4' is regel nummer 10. En de volgende regel begint op geheugenplaats 32804.

Precies deze berekening voert uw MSX-computer steeds maar weer opnieuw uit vanaf regelnummer 70, om ons bij het opzoeken van het trefwoord ook het regelnummer te kunnen opgeven.

Zo, dat was genoeg droge theorie! Laten.we nu maar eens naar 'geheugeninhoud 5' gaan kijken.

```
10 REM Geheugeninhoud 4
20 REM Karaktervolgorde in het geheugen
       zoeken
30 REM MSX Verzamelde programma's
40 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
50 SCREEN 0:COLOR 15,1,15:KEY OFF
60 WIDTH 40
70 REM Invoer van het trefwoord
80 INPUT "Welke teken(s) zoekt u
          ( niet meer dan 6 ) ";F$
90 IF LEN(F$)>6 OR LEN(F$)=0 THEN GOSUB
540:GOTO60
100 PRINT
110 REM Geheugenlocatie voor het zoeken
        kiezen
120 INPUT "In welk geheugenbereik moet n
aar          het trefwoord gezocht worden
             (/N = in het hele geheugen
              /P = alleen in het programma
) ";F1$
130 IF LEFT$(F1$,1)="p" OR LEFT$(F1$,1)=
"P" THEN F1$="P" ELSE IF LEFT$(F1$,1)="n
" OR LEFT$(F1$,1)="N" THENF1$="N"
140 PRINT
150 IF F1$="N" THEN A=0:E=65535!:GOTO 37
0
160 IF F1$="P" THEN A=32769!:GOTO 180 EL
SE GOTO 300
170 REM Zoeken naar het opgegeven tref-
        woord alleen in het programma
        als het is gevonden -> wordt de
        regelnummer aangegeven
180 AA=PEEK(A)+(PEEK(A+1)*256):BB=PEEK(A
+2)+(PEEK(A+3)*256)
190 FOR N=A TO AA-1
200 IF CHR$(PEEK(N))=MID$(F$,1,1) THEN K
= 1
```

```
210 IF K=1 AND CHR$(PEEK(N+1))=MID$(F$,2
,1) THEN K=2
220 IF K=2 AND CHR$(PEEK(N+2))=MID$(F$,3
,1) THEN K=3
230 IF K=3 AND CHR$(PEEK(N+3))=MID$(F$,4
,1) THEN K=4
240 IF K=4 AND CHR$(PEEK(N+4))=MID$(F$,5
,1) THEN K=5
250 IF K=5 AND CHR$(PEEK(N+5))=MID$(F$,6
,1) THEN K=6
260 IF K=LEN(F$) THEN PRINT F$;" IN REGE
L";BB;"GEVONDEN (";N;")"
270 K=Ø
280 NEXT N
290 A=AA:IF AA=Ø OR BB=Ø THEN GOTO 470 E
LSE GOTO180
300 PRINT
310 REM Boven- en onderwaarde begrenzen
320 INPUT "Geef het startadres op";A
330 INPUT "Geef het  eindadres op";E
340 IF A>E OR A<Ø OR E<Ø OR A>65535! OR
E>65535! THEN GOSUB 540: GOTO320
350 PRINT
360 REM Zoeken in het geheugen vanaf
        adres a t/m e naar het trefwoord
        f$
370 FOR Z=A TO E
380 IF CHR$(PEEK(Z))=MID$(F$,1,1) THEN K
=1
390 IF K=1 AND CHR$(PEEK(Z+1))=MID$(F$,2
,1) THEN K=2
400 IF K=2 AND CHR$(PEEK(Z+2))=MID$(F$,3
,1) THEN K=3
410 IF K=3 AND CHR$(PEEK(Z+3))=MID$(F$,4
,1) THEN K=4
420 IF K=4 AND CHR$(PEEK(Z+4))=MID$(F$,5
,1) THEN K=5
```

```
430 IF K=5 AND CHR$(PEEK(Z+5))=MID$(F$,6
,1) THEN K=6
440 IF K=LEN(F$) THEN PRINT F$;" op gehe
ugenplaats";Z;"gevonden"
450 K=0
460 NEXT Z
470 PRINT
480 PRINT
490 REM Programma einde of doorgaan
500 INPUT "Zoekt u nog meer trefwoorden
( /N) ";F$
510 IF LEFT$(F$,1)="n" OR LEFT$(F$,1)="N
" THEN F$="N"
520 IF F$<>"N" THEN GOTO 60
530 END
540 PRINT:PRINT TAB(10) CHR$(200) "Verke
erde opgave!" CHR$(200)
550 GOSUB 560:RETURN
560 PRINT:PRINT TAB(6) "<Druk een toets
a.u.b.>"
570 F$=INKEY$:IF F$="" THEN GOTO 570
580 PRINT:RETURN
```

```
Welk teken zoekt u
(lengte niet groter dan 6) ? lengte

In welk geheugengebied moet hiernaar
gezocht worden
( /N=in het geheugen/P=in programma) ? p

Lengte in Regel 80 gevonden ( 33054 )
```

Geheugeninhoud 5

Met het programma 'geheugenlocatie 3' hebben we destijds 8 bytes op een regel laten afbeelden. Hexadecimaal en als CHR$-waarde. In dit programma wordt alles overzichtelijker, met één byte per regel, en bovendien veelzijdiger afgebeeld.

De decimale en hexadecimale notaties worden nu naast elkaar afgebeeld:

1) Decimaal adres.
2) Hexadecimaal adres.
3) De inhoud van het adres decimaal.
4) De inhoud van het adres hexadecimaal.
5) Inhoud van het adres, indien af te beelden als CHR$.

Ten einde het geheugen te onderzoeken is dit programma bijzonder praktisch, want nu hoeft u niet steeds alles om te rekenen. Bovendien biedt dit programma een ideale mogelijkheid de veelvoud aan besturingskarakters te leren kennen. Zie tevens de appendix. Herinnert u zich nog? Een besturingskarakter is a.h.w. de afkorting waarmee een BASIC-commando in het geheugen wordt vastgelegd. Reeds eerder besproken bij geheugeninhoud 3.

We hebben een verzoek: breidt het programma eens uit met regelnummer 9. '9 PRINT' (ENTER). Zoals u weet, begint het BASIC-programma op geheugenlocatie 32769. Daarom geven we als startadres '32769' op. Het eindadres maakt niets uit. Behalve dan dat het in elk geval groter dan het startadres moet zijn. Toetsen we alleen 'ENTER' in, dan worden de adressen afgebeeld t/m het startadres + 2000 = 34796. Voor ons experiment is het echter voldoende wanneer we slechts de eerste 10 geheugenplaatsen laten afbeelden. Dus als eindadres nemen we 32779! Ten slotte, als controle, nog even regel 9 en 10 listen.

Korte verklaring van de afgebeelde geheugeninhoud.

Adres 32769:	inhoud 7	(32769 en 32770 geven het adres
Adres 32770:	inhoud 128	van de volgende regel aan:
		adr.32769+(256*adr.32770)=adr.32775)
Adres 32771:	inhoud 9	(32771 en 32772 geven het
Adres 32771:	inhoud 0	regelnummer aan
		adr.32771+(256*adr.32772)=regel nr. 9)
Adres 32773:	inhoud 145	besturingskarakter voor het commando 'PRINT'
Adres 32774:	inhoud 0	einde regelindicatie

Adres 32775:	inhoud 42	(32775 en 32776 geven het adres van de volgende regel aan.
Adres 32776:	inhoud 128	adr.32775+(256*adr.32776)=adr.32810
Adres 32777:	inhoud 10	(32777 en 32778 geven het regelnummer aan
Adres 32778:	inhoud 0	adr.32777+(256*adr.32778)=regel nr. 10
Adres 32779:	inhoud 143	besturingskarakter voor het commando 'REM'.

Voor ons is in dit verband alleen het adres 32773 interessant. Toetst u maar eens in: 'POKE 32773,143'. '143' is het besturingskarakter voor 'REM'. Als u nu regel 9 opnieuw list, zult u daar niet meer zoals eerst 'PRINT' zien staan, maar 'REM'. Er zijn niet alleen besturingskarakters van een byte zoals 'PRINT' = '145' en 'REM = '143'. Er zijn er ook die twee bytes gebruiken. Maar daar zullen we het nu maar niet over hebben. Kijkt u ter oriëntatie de appendix nog even door?

```
 32769 8001      7   07       .
 32770 8002      128 80       .
 32771 8003      9   09       .
 32772 8004      0   00       .
 32773 8005      145 91       .
 32774 8006      0   00       .
 32775 8007      42  2A       *
 32776 8008      128 80       .
 32777 8009      10  0A       .
 32778 800A      0   00       .
 32779 800B      143 8F       .
Ok


list 9-10
9 PRINT
10 REM onderzoeken van het geheugen 5
Ok
*
```

```
10 REM geheugeninhoud 5
20 REM Geheugenweergave met een byte per
       regel
30 REM MSX Verzamelde programma's
40 REM COPYRIGHT 1984 DATA BECKER &
                          Rainer lueers
50 SCREEN 0:COLOR 15,1,15:KEY OFF
60 WIDTH 40
70 REM Boven- en onderwaarde begrenzen
80 INPUT "Geef het startadres op";A
90 INPUT "Geef het  eindadres op";E
100 IF A<0 OR E<0 OR A>65535! OR E>65535
! THEN GOSUB 260:GOTO 60
110 IF E=0 THEN E=A+2000
120 CLS
130 REM Kolommen op het scherm
        adres decimaal
        adres hexadecimaal
        decimale inhoud
        hexadecimale inhoud
        weergave sla chr$(
auto 140,10
140 FOR Z=A TO E
150 PRINT Z;
160 PRINT TAB(7) STRING$(4-LEN(HEX$(Z)),
"0")+HEX$(Z);
170 PRINT TAB(17) PEEK(Z);
180 PRINT TAB(23) STRING$(2-LEN(HEX$(PEE
K(Z))),"0")+HEX$(PEEK(Z));
190 PRINT TAB(29);
200 N1=PEEK(Z)
210 IF N1<32 THEN N1=46
220 IF N1>126 THEN N1=46
230 PRINT CHR$(N1)
240 NEXT Z
250 END
260 PRINT:PRINT TAB(10) CHR$(200) " Verk
eerde opgave! " CHR$(200)
```

```
270 GOSUB 280:RETURN
280 PRINT:PRINT TAB(6) "<Druk een toets
a.u.b.>"
290 F$=INKEY$:IF F$="" THEN GOTO 290
300 RETURN
```

Grafische editor

Bij MSX-BASIC wordt zowel een grafische als een geluidsprogrammeertaal kosteloos meegeleverd. Terwijl we bij het maken van geluid met het commando 'PLAY' – bij voorbeeld 'PLAY "CDE"' – slechts naar één commando, namelijk het commando 'SOUND', kunnen verwijzen, ziet het er bij het grafisch programmeren heel wat beter uit. Zo kunnen we met de grafischeprogrammeertaal met 'DRAW' – bij voorbeeld ' DRAW "BM 128,96 R8 U8 R8" ' = teken een trap in het midden van het beeldscherm – naar een groot aantal grafische commando's verwijzen. Hetgeen ons het tekenen op het beeldscherm aanzienlijk vergemakkelijkt. We kunnen cirkels tekenen, delen van een cirkel, en zelfs ellipsen met 'CIRCLE'. Lijnen en rechthoeken met 'LINE'. En ten slotte kunnen we ook nog gesloten vlakken inkleuren met 'PAINT'. Sprites zijn met 'SPRITE$' en 'PUT SPRITE' zeer eenvoudig te construeren enz.

Alle grafische mogelijkheden worden met een oplossend vermogen van 256*192 beeldpunten in maximaal 16 kleuren op het beeldscherm weergegeven, zij het met een geringe beperking voor wat de kleurenweergave betreft (zie onder).

Het grafisch programmeren is naar verhouding eenvoudig uit te voeren met deze grote hoeveelheid aan beschikbare commando's. Maar om een tekening te maken zal men toch eerst een stuk papier, indien mogelijk millimeterpapier, nodig hebben, om later nog wijs te kunnen worden uit de doolhof van coördinaten.

Eenvoudiger werkt het met de hier afgedrukte 'grafische editor'. Door middel van het commando 'ON KEY GOSUB' wordt de status van de functietoetsen bekeken. Deze toetsen werken soms samen met de cijfertoetsen, soms met de cursortoetsen, en vaak zelfs met het hele toetsenbord en wel op de volgende manier:

F1 intoetsen: Met behulp van de cursortoetsen de radius vastleggen. (ENTER). De cirkel wordt overeenkomstig op het beeldscherm getekend.

F2 intoetsen: Met behulp van de cursortoetsen het eindpunt vastleggen. (ENTER). Er wordt een lijn getrokken tussen het start- en eindpunt.

F3 intoetsen: Twee getallen opgeven tussen 00 en 15. Vanaf nu wordt met het opgegeven kleurnummer getekend. Bij voorbeeld '15' = wit, '01' = zwart.

F4 intoetsen: Tekst in de grafiek invoeren vanaf de huidige cursorpositie, stoppen met (ENTER).

F5 intoetsen: Als voorheen met de cursor getekend werd, dan wordt vanaf nu niet meer getekend. Werd voorheen niet getekend, dan nu juist weer wel.

F6 intoetsen: Er verschijnt in het midden van de bovenste beeldschermregel een zwart veld, waarin met slechts zes letters de titel van de volgende te saven grafische voorstelling getoetst kan worden (ENTER). Nu begint het saven. Met een cassetterecorder neemt dit meer dan 10 minuten in beslag.

F7 intoetsen: Er verschijnt in het midden van de bovenste beeldschermregel een zwart veld, waarin met slechts zes letters de titel van de volgende te laden grafische voorstelling getoetst kan worden (ENTER). Nu begint het laden. Met een cassetterecorder neemt dit meer dan 10 minuten in beslag.

F8 intoetsen: Twee getallen invoeren tussen 00 en 15. Vanaf nu heeft de beeldschermrand het opgegeven kleurnummer. Bij voorbeeld '15' = wit, '01' = zwart.

F9 intoetsen: Met de op dat moment gebruikte kleur, zie F3, wordt geschilderd vanaf de cursorpositie, totdat deze kleur een rand van een gelijke kleur raakt.

F10 intoetsen: de huidige cursor is nu de linkerbenedenhoek van een beeldschermuitsnede. Een tweede cursor wordt naar de rechterbovenhoek van de beeldschermuitsnede gestuurd m.b.v. de cursortoetsen (ENTER). Een derde cursor wordt daar op het beeldscherm geplaatst waar men de linkeronderhoek van de beeldschermkopie wil hebben.

De opdrachten moeten exact volgens het bovenstaande worden ingevoerd, anders verkrijgt u een ander resultaat dan de bedoeling is.

Helaas is tijdens het tekenen niet in beeld zichtbaar met welke (functie-)toets u aan het werken bent, met uitzondering van het cursor-draadkruis. Dit is echter gedaan om uw tekenveld niet te verstoren.

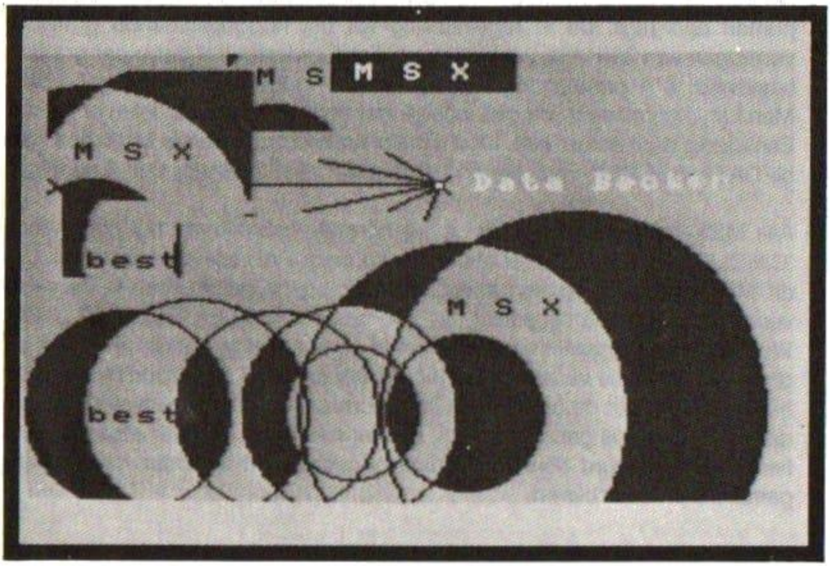
De grafische cursor bestaat uit twee afzonderlijke sprites. Het draadkruis is zwart en het draadkruiscentrum is wit. Hierdoor kan het nooit gebeuren dat het draadkruis in de tekening verdwijnt en u niet meer kunt zien waar u mee bezig bent. Maakt u gebruik van een tweede cursor, zoals bijvoorbeeld bij het vaststellen van de radius van een cirkel, dan wordt de cursor in twee delen opgedeeld. Het witte middelpunt blijft achter, terwijl het draadkruis als indicatie voor de tweede cursor wordt gebruikt. Zo weet uw MSX-computer altijd waar hij op dat ogenblik gepositioneerd wordt.

Twee beperkingen met betrekking tot de grafische mogelijkheden van de MSX-computer in het kader van dit programma verdienen nog een toelichting:

1) Er kunnen onvoorspelbare kleurreacties optreden, zoals het overlappen van kleuren, wanneer meerdere kleuren op een beeldschermregel op acht punten samenkomen. Dit verschijnsel berust heel eenvoudig op het feit dat het kleuroplossend vermogen van uw MSX-computer slechts 32*192 beeldpunten bedraagt. Dit in tegenstelling tot het hoogoplossende grafische vermogen van 256*192 beeldpunten. Hierdoor komt het dat u per grafisch beeldveld, 8*1 punten, maar twee verschillende kleuren kunt gebruiken. Men kan dit probleem wel gedeeltelijk met bepaalde hulpmiddelen omzeilen. Eenvoudig is dit echter niet. Kijkt u maar eens in mijn tweede MSX-boek dat bij DATA BECKER verschenen is, namelijk: 'MSX Grafik und Sound'.

2) Alle MSX-computers tekenen bij het normale gebruik van het commando 'CIRCLE' geen cirkel, maar een ellips op onze PAL-televisietoestellen. Om dit MSX-probleem op een zinnige manier op te lossen, heb ik de cirkel vervorming met 4/3 bijgesteld. Bij voorbeeld 'CIRCLE (128,96),20,,,,4/3)'. We hebben nu eigenlijk een commando voor een ellips gegeven. Maar op onze PAL-televisie verschijnt echter een heuse cirkel. Op grond hiervan kan het gebeuren, dat de bij het 'CIRCLE' commando gebruikte tweede cursor die voor de radius gebruikt wordt, nu niet meer precies daar staat, waar u hem heen gestuurd heeft. Na enige ervaring kunt u echter de afwijking gemakkelijk incalculeren.

Voor het overige is dit programma zeker een goede hulp. Heeft u wellicht nog een idee hoe u het verder kunt uitbreiden? Probeer het maar eens of kijk in het boek 'MSX Sound und Grafik' van DATA BECKER.



```
10 REM Grafische editor
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                          Rainer Lueers
40 REM Om te voorkomen dat men met een
       onverwachte kleurencombinatie in
       het tekstbeeldscherm terugkeert
50 ON STOP GOSUB 1970
60 STOP ON
70 REM F1 cirkel tekenen nadat de tweede
       cursor geplaatst is
   rem F2 lijn tekenen nadat de tweede
       cursor geplaatst is
   rem F3 tekenkleur veranderen twèe-
       cijferig getal intoetsen
80 REM F4 tekst vanaf de cursorpositie
       schrijven; stoppen met <ENTER>
90 REM F5 tekenen of niet tekenen niet
       tekenen is cursorbewegen
       F6 saven met naam en opschrift
       (duurt ca. 10 minuten)
       F7 inladen van een scherm
100 REM na opgave van de naam
       F8 schermrand-kleur veranderen
       altijd twee cijfers intoetsen
       (b.v. '08' maar zonder <ENTER>
110 REM F9 inkleuren met de huidige
        kleur. Voorzichtig! Er wordt ge-
        kleurd t/m eenzelfde randkleur
120 REM F10 kopie van een deel v.h.
        scherm. Huidige cursor is linker
        onderhoek. 2. Cursor=rechtsboven
        3. Cursor=linkeronderhoek v.d.
        beeldkopie
130 ON KEY GOSUB 530,660,790,830,910,940
,1170,1390,1420,1450
140 REM Voorbereidingen treffen
150 COLOR 15,4,5
```

```
160 SCREEN 2,2
170 FOR N=1 TO 10
180 KEY (N) ON
190 NEXT N
200 REM Het inlezen van het draadkruis
        wat als grafische cursor= sprite
        dient
210 FOR N=1 TO 8
220 READ A$
230 A$="&b"+A$
240 B$=B$+CHR$(VAL(A$))
250 NEXT N
260 SPRITE$(1)=B$
270 REM Het inlezen van het draadkruis
        middepunt, zodat men de cursor
        op elke ondergrond kan zien
280 FOR N=1 TO 8
290 READ A$
300 A$="&b"+A$
310 C$=C$+CHR$(VAL(A$))
320 NEXT N
330 SPRITE$(2)=C$
340 REM Het plaatsen van de het hart
        v.h. beeldscherm. Kleur zwart
350 A=128
360 B=96
370 F=1
380 COLOR F
390 PUT SPRITE 1,(A,B),1
400 PUT SPRITE 2,(A,B),15
410 REM
420 IF ME=1 THEN PSET(A+4,B+4),F
430 REM De opgegeven richting van de
        pijltoetsen bepalen
440 A$=INKEY$
450 IF A$="" THEN GOTO 440
460 IF ASC(A$)=28 THEN IF A<>250 THEN A=
A+1
```

```
470 IF ASC(A$)=29 THEN IF A<>-4 THEN A=A
-1
480 IF ASC(A$)=30 THEN IF B<>-4 THEN B=B
-1
490 IF ASC(A$)=31 THEN IF B<>186 THEN B=
B+1
500 PUT SPRITE 1,(A,B),1
510 PUT SPRITE 2,(A,B),15
520 GOTO 390
530 REM Cirkel tekenen
        ===============
540 C=A
550 D=B
560 REM De opgegeven richting van de
        pijltoetsen bepalen
570 A$=INKEY$
580 IF A$="" THEN GOTO 570
590 IF ASC(A$)=28 THEN IF C<>250 THEN C=
C+1
600 IF ASC(A$)=29 THEN IF C<>-4 THEN C=C
-1
610 IF ASC(A$)=30 THEN IF D<>-4 THEN D=D
-1
620 IF ASC(A$)=31 THEN IF D<>186 THEN D=
D+1
630 IF (A$)=CHR$(13) THEN A1=ABS(A-C):A2
=ABS(B-D):IF A1>=A2 THEN CIRCLE (A+4,B+4
),A1+13,F,,,4/3:RETURN ELSE CIRCLE (A+4,
B+4),A2,F,,,4/3:RETURN
640 PUT SPRITE 1,(C,D),1
650 GOTO 570
660 REM lijn tekenen
        ============
670 C=A
680 D=B
690 REM De opgegeven richting van de
        pijltoetsen bepalen
```

```
700 A$=INKEY$
710 IF A$="" THEN GOTO 700
720 IF ASC(A$)=28 THEN IF C<>250 THEN C=
C+1
730 IF ASC(A$)=29 THEN IF C<>-4 THEN C=C
-1
740 IF ASC(A$)=30 THEN IF D<>-4 THEN D=D
-1
750 IF ASC(A$)=31 THEN IF D<>186 THEN D=
D+1
760 IF (A$)=CHR$(13) THEN LINE (A+4,B+4)
-(C+4,D+4),F:RETURN
770 PUT SPRITE 1,(C,D),1
780 GOTO 700
790 REM Kleur veranderen
        =================
800 F$=INPUT$(2)
810 F=VAL(F$)
820 IF F<16 THEN COLOR F:RETURN ELSE RET
URN
830 REM Tekst invoeren
        ===============
840 OPEN "grp:"FOR OUTPUT AS #1
850 DRAW "bm =a;,=b;"
860 A$=INKEY$
870 IF A$="" THEN GOTO 860
880 IF A$=CHR$(13) THEN CLOSE:RETURN
890 PRINT #1,A$;
900 GOTO 860
910 REM Tekenen of niet tekenen
        ========================
920 IF ME=1 THEN ME=0 ELSE ME=1
930 RETURN
940 REM Saven
        =====
950 REM Zorgen dat het niet tot onvoor-
        spelbare kleurvermenging komen
        kan
```

```
960 LINE (100,15)-(160,0),1,BF
970 OPEN "grp:" FOR OUTPUT AS #1
980 NN$=""
990 COLOR 15
1000 DRAW "bm 108,4"
1010 REM Invoer van de programma-naam
         ofwel het beeld-opschrift met
         maximaal zes letters of cijfers
1020 FOR N=1 TO 6
1030 NN$=NN$+N$
1040 N$=INPUT$(1)
1050 IF N$=CHR$(13) THEN GOTO 1060 ELSE
PRINT #1,N$;:NEXT N
1060 A$=INKEY$
1070 REM beveiliging
1080 IF A$="" THEN GOTO 1060 ELSE IF A$<
>CHR$(13) THEN RETURN
1090 CLOSE #1
1100 OPEN NN$ FOR OUTPUT AS #1
1110 REM Saven van de 16K video-ram
1120 FOR N=0 TO 16383
1130 PRINT #1,VPEEK(N)
1140 NEXT N
1150 CLOSE
1160 RETURN
1170 REM Laden
         =====
1180 REM Zorgen dat het niet tot onvoor-
         spelbare kleurvermenging komen
         kan
1190 LINE (100,15)-(160,0),1,BF
1200 OPEN "grp:" FOR OUTPUT AS #1
1210 NN$=""
1220 COLOR 15
1230 DRAW "bm 108,4"
1240 REM Invoer van de programma-naam
         ofwel het beeld-opschrift met
         maximaal zes letters of cijfers
```

```
1250 FOR N=1 TO 6
1260 NN$=NN$+N$
1270 N$=INPUT$(1)
1280 IF N$=CHR$(13) THEN GOTO 1300 ELSE
PRINT #1,N$;:NEXT N
1290 REM beveiliging
1300 A$=INKEY$
1310 IF A$="" THEN GOTO 1300 ELSE IF A$<
>CHR$(13) THEN RETURN
1320 CLOSE #1
1330 REM laden van de 16K video-ram
1340 OPEN NN$ FOR INPUT AS #1
1350 FOR N=0 TO 16383
1360 INPUT #1,D
1370 VPOKE N,D
1380 NEXT N:CLOSE:RETURN
1390 REM rand veranderen
         ================
1400 A$=INPUT$(2)
1410 IF VAL(A$)<16 THEN COLOR ,,VAL(A$):
RETURN ELSE RETURN
1420 REM inkleuren
     ==============
1430 PAINT (A,B),F
1440 RETURN
1450 REM Het verdubbelen
         ================
1460 C1=A
1470 D1=B
1480 REM Het vaststellen van de orginele
         beeldscherm-uitsnede
         rechter bovenhoek
1490 REM De opgegeven richting van de
         pijltoetsen bepalen
1500 A$=INKEY$
1510 IF A$="" THEN GOTO 1500
1520 IF ASC(A$)=28 THEN IF C1<>250 THEN
C1=C1+1
```

```
1530 IF ASC(A$)=29 THEN IF C1<>-4 THEN C
1=C1-1
1540 IF ASC(A$)=30 THEN IF D1<>-4 THEN D
1=D1-1
1550 IF ASC(A$)=31 THEN IF D1<>186 THEN
D1=D1+1
1560 IF (A$)=CHR$(13) THEN GOTO 1600
1570 PUT SPRITE 1,(C1,D1),1
1580 GOTO 1500
1590 REM De cursor moet t.o.v. zijn
         oorsprong rechts boven worden
         gepositioneerd
1600 IF A>C1 OR B<D1 THEN BEEP:RETURN EL
SE C2=A:D2=B
1610 REM Vaststellen van de kopie
         beeldscherm-uitsnede in de
         linker benedenhoek
1620 REM De opgegeven richting van de
         pijltoetsen bepalen
1630 A$=INKEY$
1640 IF A$="" THEN GOTO 1630
1650 IF ASC(A$)=28 THEN IF C2<>250 THEN
C2=C2+1
1660 IF ASC(A$)=29 THEN IF C2<>-4 THEN C
2=C2-1
1670 IF ASC(A$)=30 THEN IF D2<>-4 THEN D
2=D2-1
1680 IF ASC(A$)=31 THEN IF D2<>186 THEN
D2=D2+1
1690 IF (A$)=CHR$(13) THEN GOTO 1730
1700 PUT SPRITE 1,(C2,D2),1
1710 GOTO 1630
1720 REM Het eigenlijke kopieeren
         uitgaande van het orgineel
1730 FOR N=A TO C1
1740 FOR M=B TO D1 STEP-1
1750 PSET(C2+(N-A),D2+(M-B)),POINT(N,M)
1760 NEXT M,N
```

```
1770 RETURN
1780 REM Sprite 1
         Het zwarte draadkruis
1790 DATA 10000001
1800 DATA 01000010
1810 DATA 00111100
1820 DATA 00100100
1830 DATA 00100100
1840 DATA 00111100
1850 DATA 01000010
1860 DATA 10000001
1870 REM Sprite 2
         Het witte draadkruis-middelpunt
1880 DATA 00000000
1890 DATA 00000000
1900 DATA 00000000
1910 DATA 00011000
1920 DATA 00011000
1930 DATA 00000000
1940 DATA 00000000
1950 DATA 00000000
1960 REM Bij een onderbreking van het
         programma wordt weer een lees-
         bare kleuropbouw bereikt
1970 COLOR 15,1
```

LISTING GRAFIKEDITOR

Sound editor

In tegenstelling tot de gezamenlijke 'toonaangevende' computersystemen, doet de firma MICROSOFT ons met hun MSX-BASIC een eenvoudige mogelijkheid aan de hand om muziek met de computer ten gehore te brengen.

Het is met het commando 'PLAY' niet alleen mogelijk de ons bekende noten in te toetsen om hierdoor de bijbehorende tonen te laten klinken. Bijvoorbeeld: 'PLAY "CDE" (ENTER)'. We kunnen zelfs driestemmige klanken te voorschijn toveren, als we drie noten d.m.v. komma's scheiden en met 'PLAY' laten spelen. Bijvoorbeeld 'PLAY "C","E","G" (ENTER)'. Wilt u de gewenste toon een octaaf hoger laten klinken? Geen probleem. U voegt gewoon de parameter "0" toe, bijvoorbeeld 'PLAY "05 CDE" (ENTER)'. Wilt u synthesizer-klanken met 'PLAY' laten horen? Dat is ook zonder meer mogelijk door het toevoegen van de parameter "S". Bijvoorbeeld 'PLAY "S8 CDE" (ENTER)'. Als u nu ook nog verdere details wilt aanbrengen, ja, dan houdt het plezier op en begint het zware werk. Steeds opnieuw moet de toon geactiveerd worden. Een getal invoeren. Een ander verwijderen. En ten slotte heeft men er geen zin meer in, omdat men het overzicht verloren heeft.

Om al onze muzikale componenten voor ons programma zonder geheugenverkwisting op te slaan en zelfs de samengestelde stemmen te mengen, is het commando 'PLAY' ideaal.

Voor het samenstellen van en het werken met gecompliceerde klankbeelden, kan men beter gebruik maken van het commando 'SOUND'. Het is abstract gezien als het ware een sprong in een muzikale getallenbrij, maar het biedt ons ook aanzienlijke voordelen:

1) Tot aan het benoemen van het golfvormpatroon blijft een met 'SOUND' opgeroepen toon klinken. Deze gaat net zo lang door totdat we hem onderbreken of een einde forceren door een foutmelding te veroorzaken.

2) Omdat we slechts eenmaal de basistoon samenstellen, kunnen we ons hierna rustig concentreren op het golfvormpatroon. We hoeven dus niet steeds alles opnieuw in te stellen.

Voor degenen onder u die de haren te berge rijzen bij het zien van de getallenbrij behorende bij het 'SOUND' commando, is deze 'sound editor' een uitstekend hulpmiddel. U wordt niet alleen het voortdurende overzicht over alle mogelijke criteria m.b.t. het veranderen van de tonen geboden, zestien in getal. U hoeft ook niet meer met de cijfertoetsen te werken, alleen nog maar met de ingebouwde pijleditor en wel op de volgende manier:

Pijl naar beneden: Naar de volgende 'SOUND' parameter overgaan.
Pijl naar boven: Naar de vorige 'SOUND' parameter.
Pijl naar links: Het getal in de bijbehorende regel met '1' verlagen.
Pijl naar rechts: Het getal in de bijbehorende regel met '1' verhogen.

En daar komt nog bij dat de editor niet alleen overzichtelijk is en daarmee gebruikersvriendelijk. Hij let er ook nog op dat er geen verkeerde waarden ingevoerd kunnen worden. Bovendien leert u bij het werken met de 'sound editor', op simpele wijze, de veelvoud aan registers van de soundprocessor een beetje beter kennen en beoordelen.

Is het u gelukt mooie driestemmige klanken voort te brengen, noteer dan de bijbehorende getallen en pas deze toe in uw eigen programma. Nadat u ook de gewenste tijdsduur, bijvoorbeeld met de commando's 'FOR', 'NEXT' en 'STEP' heeft aangepast. In een van mijn programma's uit het DATA BECKER boek 'MSX Grafik und Sound' wordt u ook dit laatste nog uit handen genomen. Bovendien komen vele andere mogelijkheden aan de orde en wordt uitvoeriger op de materie ingegaan. Probeert u het maar eens. De mogelijkheden zijn bijna onbegrensd!

Nog een tip voor het eerste gebruik van de 'sound editor'. Het is niet, zoals bij 'PLAY', voldoende om alleen de toonhoogte op te geven. Bijvoorbeeld 'PLAY "C"'. Om een toon weer te kunnen geven moet ten minste het onderstaande ingesteld worden:

1) De toonhoogte van het betreffende kanaal minstens op '1'.
2) Het kanaal op 'Toon = 1', niet op ruis.
3) Het volume van het betreffende kanaal minstens op '1'.

Hierna kunt u uw fantasie de vrije loop laten en naar hartelust gaan experimenteren.

```
                Sound editor
                ============

     toonhoogte fijn kanaal 1:   21
     toonhoogte grof kanaal 1:   5
     toonhoogte fijn kanaal 2:   0
     toonhoogte grof kanaal 2:   0
     toonhoogte fijn kanaal 3:   108
     toonhoogte grof kanaal 3:   2

         kanaal 1: 0/1=t/2=r:    1
         kanaal 2: 0/1=t/2=r:    2
         kanaal 3: 0/1=t/2=r:    1

              volume kanaal 1:   11
              volume kanaal 2:   16
              volume kanaal 3:   14

                  ruisperiode:   10
         golfvormpatroon fijn:   5
         golfvormpatroon grof:   7
                     golfvorm:   11
```

```
10 REM Sound editor
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                          Rainer Lueers
40 REM Voorbereidingen treffen
50 SCREEN 0:WIDTH 35
60 CLEAR 2000:DIM A$(21),A(21)
70 COLOR 15,1,1
80 ON STOP GOSUB 1130:STOP ON
90 REM Alle soundparameters in de
       uitgangspositie zetten
100 FOR N=0 TO 13
110 SOUND N,0
120 NEXT N
130 SOUND 7,63
140 REM Opbouw van het beeldscherm
150 PRINT TAB(7) "Sound editor"
160 PRINT TAB(7) "============"
170 PRINT
180 PRINT TAB(1);:A$(3)="toonhoogte fijn
 kanaal 1:":PRINT A$(3)
190 PRINT TAB(1);:A$(4)="toonhoogte grof
 kanaal 1:":PRINT A$(4)
200 PRINT TAB(1);:A$(5)="toonhoogte fijn
 kanaal 2:":PRINT A$(5)
210 PRINT TAB(1);:A$(6)="toonhoogte grof
 kanaal 2:":PRINT A$(6)
220 PRINT TAB(1);:A$(7)="toonhoogte fijn
 kanaal 3:":PRINT A$(7)
230 PRINT TAB(1);:A$(8)="toonhoogte grof
 kanaal 3:":PRINT A$(8)
240 PRINT
250 PRINT TAB(6);:A$(10)="kanaal 1: 0/1=
t/2=r:":PRINT A$(10)
260 PRINT TAB(6);:A$(11)="kanaal 2: 0/1=
t/2=r:":PRINT A$(11)
270 PRINT TAB(6);:A$(12)="kanaal 3: 0/1=
t/2=r:":PRINT A$(12)
```

```
280 PRINT
290 PRINT TAB(10);:A$(14)="volume kanaal
 1:":PRINT A$(14)
300 PRINT TAB(10);:A$(15)="volume kanaal
 2:":PRINT A$(15)
310 PRINT TAB(10);:A$(16)="volume kanaal
 3:":PRINT A$(16)
320 PRINT
330 PRINT TAB(14);:A$(18)="ruisperiode:"
:PRINT A$(18)
340 PRINT TAB(5);:A$(19)="golfvormpatroo
n fijn:":PRINT A$(19)
350 PRINT TAB(5);:A$(20)="golfvormpatroo
n grof:":PRINT A$(20)
360 PRINT TAB(17);:A$(21)="golfvorm:":PR
INT A$(21)
370 REM Waardeaanduiding van de para-
        meters op het beeldscherm
380 FOR N=3 TO 8
390 LOCATE 26,N
400 PRINT A
410 NEXT N
420 FOR N=10 TO 12
430 LOCATE 26,N
440 PRINT A
450 NEXT N
460 FOR N=14 TO 16
470 LOCATE 26,N
480 PRINT A
490 NEXT N
500 FOR N=18 TO 21
510 LOCATE 26,N
520 PRINTA
530 NEXT N
540 REM Het plaatsen van de cursors in
        de juiste beeldschermregel op
        positie 0
```

```
550 LOCATE Ø,3:PRINT CHR$(2ØØ)
56Ø REM Op invoer wachten; deze kan
        bestaan uit cursortoetsen m.a.w.
        pijl naar boven of pijl naar
        beneden -> cursorpositionering
57Ø REM Pijl naar links of pijl naar
        rechts -> parameters +/-
58Ø A$=INKEY$:IF A$="" THEN GOTO58Ø
59Ø IF A$=" " THEN SOUND 13,A(21):GOTO 5
8Ø
6ØØ IF ASC(A$)=3Ø OR ASC(A$)=31 THEN GOT
O 61Ø ELSE IF ASC(A$)=28 OR ASC(A$)=29 T
HEN GOTO 69Ø ELSE GOTO 58Ø
61Ø IF ASC(A$)=3Ø AND CSRLIN=3 THEN GOTO
 58Ø
62Ø IF ASC(A$)=31 AND CSRLIN=21 THEN GOT
O 58Ø
63Ø IF ASC(A$)=31 THEN IF CSRLIN=8 OR CS
RLIN=12 OR CSRLIN=16 THEN A=2 ELSE A=1 E
LSE GOTO 66Ø
64Ø LOCATE Ø,CSRLIN:PRINT " ";:LOCATE Ø,
CSRLIN+A:PRINT CHR$(2ØØ);:LOCATE 25,CSRL
IN
65Ø GOTO 58Ø
66Ø IF ASC(A$)=3Ø THEN IF CSRLIN=18 OR C
SRLIN=14 OR CSRLIN=1Ø THEN A=-2 ELSE A=-
1 ELSE GOTO 58Ø
67Ø LOCATE Ø,CSRLIN:PRINT " ";:LOCATE Ø,
CSRLIN+A:PRINT CHR$(2ØØ);:LOCATE 25,CSRL
IN
68Ø GOTO 58Ø
69Ø IF ASC(A$)=28 THEN Z=1 ELSE Z=-1
7ØØ LOCATE 26,CSRLIN:PRINT "    ";:LOCATE
 26,CSRLIN
71Ø REM Invoeren van de parameters
        + of -  in het 'SOUND'-commando;
        bovendien controleren van de
        grenswaarden en de beeldscherm-
        weergave
```

```
720 REM Toonhoogte kanaal 1 fijn
        ==========================
730 IF CSRLIN=3 THEN IF (A(3)+Z<Ø OR A(3
)+Z>255) THEN PRINT A(3);:GOTO 58Ø ELSE
A(3)=A(3)+Z:PRINT A(3);:SOUND Ø,A(3):GOT
O 58Ø
740 REM Toonhoogte kanaal 1 grof
        ==========================
750 IF CSRLIN=4 THEN IF (A(4)+Z<Ø OR A(4
)+Z>15) THEN PRINT A(4);:GOTO 58Ø ELSE A
(4)=A(4)+Z:PRINT A(4);:SOUND 1,A(4):GOTO
 58Ø
760 REM Toonhoogte kanaal 2 fijn
        ==========================
770 IF CSRLIN=5 THEN IF (A(5)+Z<Ø OR A(5
)+Z>255) THEN PRINT A(5);:GOTO 58Ø ELSE
A(5)=A(5)+Z:PRINT A(5);:SOUND 2,A(5):GOT
O 58Ø
780 REM Toonhoogte kanaal 2 grof
        ==========================
790 IF CSRLIN=6 THEN IF (A(6)+Z<Ø OR A(6
)+Z>15) THEN PRINT A(6);:GOTO 58Ø ELSE A
(6)=A(6)+Z:PRINT A(6);:SOUND 3,A(6):GOTO
 58Ø
8ØØ REM Toonhoogte kanaal 3 fijn
        ==========================
81Ø IF CSRLIN=7 THEN IF (A(7)+Z<Ø OR A(7
)+Z>255) THEN PRINT A(7);:GOTO 58Ø ELSE
A(7)=A(7)+Z:PRINT A(7);:SOUND 4,A(7):GOT
O 58Ø
82Ø REM Toonhoogte kanaal 3 grof
        ==========================
83Ø IF CSRLIN=8 THEN IF (A(8)+Z<Ø OR A(8
)+Z>15) THEN PRINT A(8);:GOTO 58Ø ELSE A
(8)=A(8)+Z:PRINT A(8);:SOUND 5,A(8):GOTO
 58Ø
84Ø REM Volume kanaal 1
        ================
```

```
850 IF CSRLIN=14 THEN IF (A(14)+Z<Ø OR A
(14)+Z>16) THEN PRINT A(14);:GOTO 58Ø EL
SE A(14)=A(14)+Z:PRINT A(14);:SOUND 8,A(
14):GOTO 58Ø
86Ø REM Volume kanaal 2
        =================
87Ø IF CSRLIN=15 THEN IF (A(15)+Z<Ø OR A
(15)+Z>16) THEN PRINT A(15);:GOTO 58Ø EL
SE A(15)=A(15)+Z:PRINT A(15);:SOUND 9,A(
15):GOTO 58Ø
88Ø REM Volume kanaal 3
        =================
89Ø IF CSRLIN=16 THEN IF (A(16)+Z<Ø OR A
(16)+Z>16) THEN PRINT A(16);:GOTO 58Ø EL
SE A(16)=A(16)+Z:PRINT A(16);:SOUND 1Ø,A
(16):GOTO 58Ø
9ØØ REM Ruisperiode
    ================
91Ø IF CSRLIN=18 THEN IF (A(18)+Z<Ø OR A
(18)+Z>31) THEN PRINT A(18);:GOTO 58Ø EL
SE A(18)=A(18)+Z:PRINT A(18);:SOUND 6,A(
18):GOTO 58Ø
92Ø REM golfvormpatroon fijn
        ======================
93Ø IF CSRLIN=19 THEN IF (A(19)+Z<Ø OR A
(19)+Z>255) THEN PRINT A(19);:GOTO 58Ø E
LSE A(19)=A(19)+Z:PRINT A(19);:SOUND 11,
A(19):GOTO 58Ø
94Ø REM golfvormpatroon grof
        ======================
95Ø IF CSRLIN=2Ø THEN IF (A(2Ø)+Z<Ø OR A
(2Ø)+Z>255) THEN PRINT A(2Ø);:GOTO 58Ø E
LSE A(2Ø)=A(2Ø)+Z:PRINT A(2Ø);:SOUND 12,
A(2Ø):GOTO 58Ø
96Ø REM golfvorm
        =========
```

```
970 IF CSRLIN=21 THEN IF (A(21)+Z<Ø OR A
(21)+Z>15) THEN PRINT A(21);:GOTO 58Ø EL
SE A(21)=A(21)+Z:PRINT A(21);:SOUND 13,A
(21):GOTO 58Ø
98Ø REM Kanaal 1:uit/toon/ruis/beide
        ================================
99Ø IF CSRLIN=1Ø THEN IF (A(1Ø)+Z<Ø OR A
(1Ø)+Z>3) THEN PRINT A(1Ø);:GOTO 58Ø ELS
E A(1Ø)=A(1Ø)+Z:PRINT A(1Ø);
1ØØØ REM Kanaal 2:uit/toon/ruis/beide
         ================================
1Ø1Ø IF CSRLIN=11 THEN IF (A(11)+Z<Ø OR
A(11)+Z>3) THEN PRINT A(11);:GOTO 58Ø EL
SEA(11)=A(11)+Z:PRINT A(11);
1Ø2Ø REM Kanaal 3:uit/toon/ruis/beide
         ================================
1Ø3Ø IF CSRLIN=12 THEN IF (A(12)+Z<Ø OR
A(12)+Z>3) THEN PRINT A(12);:GOTO 58Ø EL
SE A(12)=A(12)+Z:PRINT A(12);
1Ø4Ø IF CSRLIN<1Ø OR CSRLIN>12 THEN GOTO
 112Ø
1Ø5Ø REM Precieze berekening van de
          'SOUND'-parameters 7
1Ø6Ø A=Ø
1Ø7Ø IF A(1Ø)=1 THEN A=A+8 ELSE IF A(1Ø)
=2 THEN A=A+1 ELSE IF A(1Ø)=Ø THEN A=A+1
+8
1Ø8Ø IF A(11)=1 THEN A=A+16 ELSE IF A(11
)=2 THEN A=A+2 ELSE IF A(11)=Ø THEN A=A+
16+2
1Ø9Ø IF A(12)=1 THEN A=A+32 ELSE IF A(12
)=2 THEN A=A+4 ELSE IF A(12)=Ø THEN A=A+
32+4
11ØØ A=A+128
111Ø SOUND 7,A
112Ø GOTO 58Ø
113Ø WIDTH 4Ø
```

Duitse karakterset

Iedereen moppert er wel eens over als hij een moderne MSX-computer naast de ouwe getrouwe schrijfmachine ziet staan. Aan de ene kant een internationale computer, natuurlijk ook uitgerust met een internationale karakterset, dus zonder umlaut en een sz. Aan de andere kant de schrijfmachine, die wel beschikt over al die nuttige zaken, maar die daarentegen, normaal gesproken, geen computer is.

Met dit programma maken we van uw MSX-computer een tegenhanger van de schrijfmachine, tenminste wat betreft de beeldschermweergave. De 'z' zit weliswaar niet daar waar hij zitten moet, namelijk bij de 'y' en omgekeerd. Ook de umlauten 'ae', 'ue', 'oe' evenals de 'sz' zitten nog niet op die plaats van het toetsenbord waar ze eigenlijk hadden moeten zitten. Maar we kunnen heel gemakkelijk van deze Duitse karakterset gebruik maken en niet alleen op het beeldscherm zichtbaar maken, maar zelfs in combinatie met e l k e printer gebruiken.

Zoals u zeker al bekend zal zijn, beschikt de MSX-computer, die voor de Europese markt gefabriceerd is, reeds over Europese karakters. Wat de Duitse markt betreft:

'sz' m.b.v. 'CODE' en '7'
'ae' m.b.v. 'CODE' en 'a'
'Ae' m.b.v. 'SHIFT','CODE' en 'a'
'oe' m.b.v. 'CODE' en 'f'
'Oe' m.b.v. 'SHIFT','CODE' en 'f'
'ue' m.b.v. 'CODE' en 'g'
'Ue' m.b.v. 'SHIFT','CODE' en 'g'

Hoewel deze bediening van de toetsen nogal ongebruikelijk is, verkrijgt u in elk geval wel het gewenste resultaat op het beeldscherm. Op een printer werkt het alleen maar als u gebruik maakt van een speciale MSX-printer, terwijl het hierna afgedrukte programma ook met alle normale printers werkt, zoals bijvoorbeeld een Epson printer.

Het programma start u met 'RUN'. Hierna wordt u gevraagd met welk scherm u werken wilt. Het antwoord op deze vraag is belangrijk, want door het wisselen van scherm met het commando 'SCREEN' wist u de aangepaste karakterset en moet het programma 'Duitse karakterset' opnieuw geladen worden.

Hierna wordt u gevraagd of uw computer reeds over de voor de Europese markt bruikbare karakterset beschikt. Probeert u dat maar even uit. Als u de toetsen zoals eerder aangegeven indrukt, verschijnen de bijbehorende karakters. Als dit het geval is, kunt u de vraag met 'j'(a) beantwoorden. Uw MSX-computer zal nu de umlauten uit de reeds bestaande karakterset halen en plaatsen onder de toetsen zoals hierna afgebeeld. Is dit niet het geval dan bezit u mogelijk een MSX-computer die voor de Japanse markt bedoeld is. In dat geval worden de umlauten uit de DATA regels gehaald en alsnog onder genoemde toetsen geplaatst.

We kunnen nu de 'Duitse karakterset' gebruiken en hiermee ook een afdruk op een normale printer met een centronics aansluiting verkrijgen.

'[' CHR$(91) = 'A'
'{' CHR$(123) = 'a'
']' CHR$(93) = 'U'
'}' CHR$(125) = 'u'
'~' CHR$(126) = ' '
'\' CHR$(92) = 'O'
'¦' CHR$(124) = 'o'

Toelichting vertaler

In eerste instantie zult u zich afvragen wat voor u het nut zou kunnen zijn als u de beschikking heeft over een Duitse karakterset. Welnu, u heeft dit boek aangeschaft om wat te leren, nietwaar? Wat dit laatste betreft, biedt dit programma toch wel een aantal leuke mogelijkheden. 'De praktijk is de beste leermeester' is misschien een oud gezegde, maar gaat wat betreft het leren programmeren nog steeds op. In feite is dit de enige manier om op dit gebied kennis te verwerven. Laten we daarom het ogenschijnlijk alleen voor Duitse markt geschikte programma eens nader bekijken. Vanaf regel 420 vindt u een aantal DATA regels. Als we ons als voorbeeld nu even concentreren op regel 420 t/m regel 500. Dan zult u bemerken dat de cijfers één in deze regels gezamenlijk de hoofdletter A met een umlaut vormen. Wat zou u er dan van weerhouden hiervan uw eigen karakterteken te maken? Wat dacht u van een guldenteken of een 'E' met dubbele punt? Gewoon de cijfers één veranderen in een patroon dat u afgebeeld wilt hebben. Op deze manier kunt u dan totaal 7 eigen karakters maken en gebruiken. Zoals reeds eerder vermeld worden deze DATA regels alleen ingelezen als u een Japanse uitvoering bezit i.p.v. een Europese. Nadat u het programma met 'RUN' gestart heeft, dient u dus te antwoorden alsof u over een Japanse uitvoering beschikt.

```
10 REM Duitse karakterset
20 REM MSX Verzamelde programma's
30 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
40 REM Het initialiseren van
       SCREEN 0 en 1
50 SCREEN 1:WIDTH 29:SCREEN 0:WIDTH 40:C
OLOR 15,1:KEY OFF
60 REM Voorbereidingen treffen
70 CLS:INPUT "Screen (0/1) ";SC
80 REM Aanwenden van SCREEN 1 of
                         SCREEN 0
90 INPUT "Duitse karakterset beschikbaar
 (J/ ) ";F$:IF LEFT$(F$,1)="J" OR LEFT$(
F$,1)="j" THEN F$="J"
100 IF SC=0 THEN SCREEN 0:RESTORE 240 EL
SE SCREEN 1:RESTORE 320
110 REM Programadeel voor MSX-computers
        waarvan de karakterset niet aan
        de Europese norm voldoet.
        Letters met dubbele punt wel
        aanwezig maar op de verkeerde
        plaats
120 IF F$<>"J" THEN FOR N=1 TO 7:READ A(
N),A:NEXT N:RESTORE 430:FOR N=1 TO 7:FOR
 M=0 TO 7:READ A$:A$="&b"+A$:VPOKE A(N)+
M,VAL(A$):NEXT M,N:GOTO 210
130 REM Hoofdprogramma voor MSX-computers
        die aan de Europese norm voldoen
        M.a.w. de dubbele punten zijn
        weliswaar aanwezig, maar de
        weergave op het beeld komt in
        het geheel niet overeen
140 REM De weergave op de printer
150 FOR N=1 TO 7
160 READ A,B
170 FOR M=0 TO 7
180 VPOKE A+M,VPEEK(B+M)
```

```
190 NEXT M,N
200 REM demonstratie-tekst
        ==================
210 IF SC=0 THEN PRINT "£ber den groaen
Flua gehen zu d£rfen,   das bedeutet f£r
 mich ein rgernis      ohnegleichen ...
ich rufe laut 'R⌊MER'  ... mmh ... ein l
biches Wort auf ...    ``h ... jetzt f`ll
t mir nichts mehr ein."
220 IF SC=1 THEN PRINT "£ber den groaen
Flua zud£rfen, das bedeutet f£r mich
  ein rgernis ohnegleichen     ... ich ru
fe laut 'R⌊mer' ...mmh ... ein lbliches
Wort   auf ... ``h ... jetzt f`llt  mir
nichts mehr ein."
230 REM Als er dubbele punten zijn->
         SCREEN 0
240 DATA 2776,3184:REM 'AE'
250 DATA 3032,3104:REM 'ae'
260 DATA 2792,3280:REM 'UE'
270 DATA 3048,3080:REM 'ue'
280 DATA 3056,3848:REM 'sz'
290 DATA 2784,3272:REM 'OE'
300 DATA 3040,3232:REM 'oe'
310 REM Als er dubbele punten zijn->
        SCREEN 1
320 DATA 728,1136:REM 'AE'
330 DATA 984,1056:REM 'ae'
340 DATA 744,1232:REM 'UE'
350 DATA 1000,1032:REM 'ue'
360 DATA 1008,1800:REM 'sz'
370 DATA 736,1224:REM 'OE'
380 DATA 992,1184:REM 'oe'
390 REM datalijst voor dubbele punten
        en de 'sz'
400 REM ============================
410 REM De letter 'AE'
420 REM chr$(91)
```

```
430 DATA 01010000
440 DATA 00000000
450 DATA 00100000
460 DATA 01010000
470 DATA 10001000
480 DATA 11111000
490 DATA 10001000
500 DATA 00000000
510 REM De letter 'ae'
520 REM chr$(123)
530 DATA 01001000
540 DATA 00000000
550 DATA 01110000
560 DATA 00001000
570 DATA 01111000
580 DATA 10001000
590 DATA 01111000
600 DATA 00000000
610 REM De letter 'UE'
620 REM chr$(93)
630 DATA 01010000
640 DATA 00000000
650 DATA 10001000
660 DATA 10001000
670 DATA 10001000
680 DATA 10001000
690 DATA 01110000
700 DATA 00000000
710 REM De letter 'ue'
720 REM chr$(125)
730 DATA 10010000
740 DATA 00000000
750 DATA 00000000
760 DATA 10010000
770 DATA 10010000
780 DATA 10010000
790 DATA 01101000
800 DATA 00000000
```

```
810 REM De letter 'sz'
820 REM chr$(126)
830 DATA 00110000
840 DATA 01001000
850 DATA 01001000
860 DATA 01110000
870 DATA 01001000
880 DATA 01001000
890 DATA 01110000
900 DATA 11000000
910 REM De letter 'OE'
920 REM chr$(92)
930 DATA 01010000
940 DATA 00000000
950 DATA 01110000
960 DATA 10001000
970 DATA 10001000
980 DATA 10001000
990 DATA 01110000
1000 DATA 00000000
1010 REM De letter 'oe'
1020 REM chr$(124)
1030 DATA 10010000
1040 DATA 00000000
1050 DATA 00000000
1060 DATA 01100000
1070 DATA 10010000
1080 DATA 10010000
1090 DATA 01100000
1100 DATA 00000000
```

über den grossen Fluss gehen zu dürfen,
das bedeutet für mich ein Ärgernis
ohnegleichen ... ich rufe laut 'Römer'
... mmh ... ein löbliches Wort auf ...
ääh ... jetzt fällt mir nichts mehr ein.

Ok

*

Computerkarakterset

Eerlijk gezegd kan men stellen, dat de karakterset van uw MSX-computer zeer omvangrijk is. Zo kunnen we niet alleen beschikken over de letters 'A' t/m 'Z' in kleine en hoofdletters, maar we hebben ook de beschikking over wiskundige tekens en vele grafische tekens waar we een boel plezier aan kunnen beleven.

Waarom dan ook nog een andere karakterset? Al jaren lang houdt de euforie van computerspellen aan. Waarom zouden we voor zoiets futuristisch een normale karakterset gebruiken? Hier is dan de bijbehorende computerkarakterset!

Het programma bestaat uit twee hoofdmoten:

1) Een nagenoeg onoverzienbare reeks DATA commando's met de computerkarakterset.
2) De verwerkings- en laadroutine voor deze DATA.

Uw MSX-computer kan op het tekstbeeldscherm maximaal 256 verschillende karakters 'CHR$' tegelijkertijd afbeelden. Als het uiterlijk van een karakter wordt veranderd, veranderen tegelijkertijd alle overeenkomstige karakters op het beeldscherm.

Met dit programma kunt u het volgende bereiken. U kunt ofwel de complete karakterset in computerschrift veranderen, of met normaal en computerschrift gecombineerd werken.

Door het "ompoken" van het video-RAM-geheugen verkrijgen we de complete opnieuw samengestelde karakterset.

Hoe kunnen we nu twee karaktersets tegelijkertijd in het geheugen opslaan? Hiervoor wordt de normale karakterset vanaf CHR$(0) t/m CHR$(255) vanaf 128 veranderd. Met andere woorden: we verliezen tijdens het gebruik van dit programma een groot deel van de extra karakters zoals de grafische en wiskundige tekens. We krijgen hiervoor in de plaats deze computerkarakterset. Om vanuit het programma te kunnen omschakelen van de ene op de andere karakterset, hebben we als het ware een schakelaar tussen de verschillende karaktersets ingebouwd. '=)' indrukken -) computerkarakterset aan. '=)' opnieuw indrukken -) normale karakterset aan.

Programmatechnisch gezien gebeurt het volgende: de ASCII-code van het gewenste karakter wordt bekeken; bijvoorbeeld '!' = 33, hier wordt het bijbehorende computerkarakter gezocht en op het beeldscherm afgebeeld.

```
Uw keuze alstublieft?
Wilt u computerschrift?
Druk dan
Wilt u normaal-schrift?
druk dan
Wilt u alles in computerschrift
ook de listing?
Toets dan J

Geef de tekst in a.u.b
```

```
10 REM Computerkarakterset
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                     Rainer Lueers
40 SCREEN1:COLOR15,1,1:KEY OFF
50 WIDTH 30
60 FOR M=129 TO 218:PRINT CHR$(M);:NEXT:
PRINT
70 REM Inlezen van het computerschrift
       in de karakters boven de 126
80 Z=31:FOR N=1 TO 73
90 READ A:Z=Z+1
100 IF Z=A THEN FOR M=1 TO 8 ELSE FOR M=
1 TO 8:VPOKE 1023+M+((Z-32)*8),VPEEK(255
+M+((Z-32)*8)):NEXT M:Z=Z+1:GOTO 100
110 FOR M=1 TO 8
120 READ A$
130 REM Het veranderen van de string-
        weergave in een binair getal en
        hieropvolgend in een decimaal
        getal
140 A$="&b"+A$
150 VPOKE 1031+M+((Z-33)*8),VAL(A$)
160 NEXT M
170 NEXT N
180 REM Keuze-mogelijkheden
        1) tijdens het programma text
           in computerschrift of normaal
           schrift invoeren
190 REM 2) de gehele normale karakterset
           van chr$(32) t/m chr$(126) in
           computerschrift veranderen
200 PRINT "Uw keuze alstublieft ?
 Wilt u computerschrift?         Druk dan
op ->                  Wilt u normaal-schr
ift?       Druk dan op <-
210 PRINT "Wilt u alles in computer-
 schrift ook de listing?         Toets dan
 J";:A$="":INPUT A$
```

```
220 IF LEFT$(A$,1)="J" OR LEFT$(A$,1)="j
" THEN A$="J"
230 IF A$="J" THEN GOTO 510
240 REM De normale karakterset blijft
        behouden slechts enkele letters
        cijfers en tekens worden indien
        gewenst in computerkarakters
        veranderd
250 PRINT
260 PRINT "Geef de tekst in a.u.b."
270 PRINT
280 A$=INKEY$
290 IF A$="" THEN GOTO 280
300 REM Kleine tekst editor: De toetsen
        <ENTER> =13 en -> =29 en <- =28
        worden bekeken
310 IF ASC(A$)=28 THEN GOTO 370
320 IF ASC(A$)=13 THEN PRINT:GOTO 280
330 IF ASC(A$)=29 THEN A$="":IF POS(0)>0
 THEN LOCATE POS(0)-1,CSRLIN:PRINT " ";:
LOCATE POS(0)-1,CSRLIN:GOTO 280
340 PRINT A$;
350 GOTO 280
360 END
370 A$=INKEY$
380 IF A$="" THEN GOTO 370
390 REM Kleine tekst editor: De toetsen
        <ENTER> =13 en -> =29 en <- =28
        worden bekeken
400 IF ASC(A$)=29 THEN GOTO 280
410 IF ASC(A$)=13 THEN PRINT:GOTO 370
420 IF ASC(A$)=28 THEN A$="":IF POS(0)>0
 THEN LOCATE POS(0)-1,CSRLIN:PRINT " ";:
LOCATE POS(0)-1,CSRLIN:GOTO 370 ELSE GOT
O 370
430 IF ASC(A$)>122 THEN PRINT A$;:GOTO 3
70
```

```
440 REM Elk karakter wordt uit de be-
        schikbare karakterstring gelezen
        zodat de normale karakterset
        niet wordt verstoord
450 Z=ASC(A$)
460 REM Het kiezen van het karakter
470 PRINT CHR$(Z+96);
480 GOTO 370
490 END
500 REM Het inlezen van de computer-
        karakters en het veranderen van
        de normale karakterset
510 FOR N=0 TO 719
520 VPOKE 264+N,VPEEK(1032+N)
530 NEXT N
540 END
550 REM !
560 DATA 33
570 DATA 00111000
580 DATA 00111000
590 DATA 00111000
600 DATA 00111000
610 DATA 00011000
620 DATA 00000000
630 DATA 00011000
640 DATA 00000000
650 REM "
660 DATA 34
670 DATA 01100110
680 DATA 01100110
690 DATA 01100110
700 DATA 00000000
710 DATA 00000000
720 DATA 00000000
730 DATA 00000000
740 DATA 00000000
750 REM #
760 DATA  35
770 DATA 01100110
```

```
780 DATA 11111111
790 DATA Ø11ØØ11Ø
800 DATA Ø11ØØ11Ø
810 DATA 11111111
820 DATA Ø11ØØ11Ø
830 DATA ØØØØØØØØ
840 DATA ØØØØØØØØ
850 REM '
860 DATA 39
870 DATA ØØØ11ØØØ
880 DATA ØØØ11ØØØ
890 DATA ØØØ11ØØØ
900 DATA ØØØØØØØØ
910 DATA ØØØØØØØØ
920 DATA ØØØØØØØØ
930 DATA ØØØØØØØØ
940 DATA ØØØØØØØØ
950 REM (
960 DATA 40
970 DATA ØØØ1111Ø
980 DATA ØØØ11ØØØ
990 DATA ØØØ11ØØØ
1000 DATA ØØ111ØØØ
1010 DATA ØØ111ØØØ
1020 DATA ØØ111ØØØ
1030 DATA ØØ11111Ø
1040 DATA ØØØØØØØØ
1050 REM )
1060 DATA 41
1070 DATA Ø1111ØØØ
1080 DATA ØØØ11ØØØ
1090 DATA ØØØ11ØØØ
1100 DATA ØØØ111ØØ
1110 DATA ØØØ111ØØ
1120 DATA ØØØ111ØØ
1130 DATA Ø11111ØØ
1140 DATA ØØØØØØØØ
1150 REM Ø
1160 DATA 48
```

```
1170 DATA 01111111
1180 DATA 01100011
1190 DATA 01100011
1200 DATA 01100011
1210 DATA 01100011
1220 DATA 01100011
1230 DATA 01111111
1240 DATA 00000000
1250 REM 1
1260 DATA 49
1270 DATA 00111000
1280 DATA 00011000
1290 DATA 00011000
1300 DATA 00011000
1310 DATA 00111110
1320 DATA 00111110
1330 DATA 00111110
1340 DATA 00000000
1350 REM 2
1360 DATA 50
1370 DATA 01111111
1380 DATA 00000011
1390 DATA 00000011
1400 DATA 01111111
1410 DATA 01100000
1420 DATA 01100000
1430 DATA 01111111
1440 DATA 00000000
1450 REM 3
1460 DATA 51
1470 DATA 01111110
1480 DATA 00000110
1490 DATA 00000110
1500 DATA 01111111
1510 DATA 00000111
1520 DATA 00000111
1530 DATA 11111111
1540 DATA 00000000
1550 REM 4
```

```
156Ø DATA 52
157Ø DATA Ø111ØØØØ
158Ø DATA Ø111ØØØØ
159Ø DATA Ø111ØØØØ
16ØØ DATA Ø111Ø111
161Ø DATA Ø111Ø111
162Ø DATA Ø1111111
163Ø DATA ØØØØØ111
164Ø DATA ØØØØØØØØ
165Ø REM 5
166Ø DATA 53
167Ø DATA Ø1111111
168Ø DATA Ø11ØØØØØ
169Ø DATA Ø11ØØØØØ
17ØØ DATA Ø1111111
171Ø DATA ØØØØØ111
172Ø DATA ØØØØØ111
173Ø DATA Ø1111111
174Ø DATA ØØØØØØØØ
175Ø REM 6
176Ø DATA 54
177Ø DATA Ø11111ØØ
178Ø DATA Ø11Ø11ØØ
179Ø DATA Ø11ØØØØØ
18ØØ DATA Ø1111111
181Ø DATA Ø11ØØØ11
182Ø DATA Ø11ØØØ11
183Ø DATA Ø1111111
184Ø DATA ØØØØØØØØ
185Ø REM 7
186Ø DATA 55
187Ø DATA Ø1111111
188Ø DATA ØØØØØØ11
189Ø DATA ØØØØØØ11
19ØØ DATA ØØØ11111
191Ø DATA ØØØ11ØØØ
192Ø DATA ØØØ11ØØØ
193Ø DATA ØØØ11ØØØ
194Ø DATA ØØØØØØØØ
```

```
1950 REM 8
1960 DATA 56
1970 DATA 00111110
1980 DATA 00110110
1990 DATA 00110110
2000 DATA 01111111
2010 DATA 01110111
2020 DATA 01110111
2030 DATA 01111111
2040 DATA 00000000
2050 REM 9
2060 DATA 57
2070 DATA 01111111
2080 DATA 01100011
2090 DATA 01100011
2100 DATA 01111111
2110 DATA 00000111
2120 DATA 00000111
2130 DATA 00000111
2140 DATA 00000000
2150 REM :
2160 DATA 58
2170 DATA 00000000
2180 DATA 00011000
2190 DATA 00011000
2200 DATA 00000000
2210 DATA 00011000
2220 DATA 00011000
2230 DATA 00000000
2240 DATA 00000000
2250 REM ;
2260 DATA 59
2270 DATA 00000000
2280 DATA 00011000
2290 DATA 00011000
2300 DATA 00000000
2310 DATA 00011000
2320 DATA 00011000
2330 DATA 00110000
2340 DATA 00000000
```

```
2350 REM =
2360 DATA 61
2370 DATA 00000000
2380 DATA 01111110
2390 DATA 00000000
2400 DATA 00000000
2410 DATA 01111110
2420 DATA 00000000
2430 DATA 00000000
2440 DATA 00000000
2450 REM ?
2460 DATA 63
2470 DATA 01111111
2480 DATA 01100011
2490 DATA 00000011
2500 DATA 00011111
2510 DATA 00011100
2520 DATA 00000000
2530 DATA 00011100
2540 DATA 00000000
2550 REM @
2560 DATA 64
2570 DATA 01111111
2580 DATA 01100011
2590 DATA 01101111
2600 DATA 01101111
2610 DATA 01101111
2620 DATA 01100000
2630 DATA 01111111
2640 DATA 00000000
2650 REM A
2660 DATA 65
2670 DATA 00111111
2680 DATA 00110011
2690 DATA 00110011
2700 DATA 01111111
2710 DATA 01110011
2720 DATA 01110011
2730 DATA 01110011
2740 DATA 00000000
```

```
2750 REM B
2760 DATA 66
2770 DATA 01111110
2780 DATA 01100110
2790 DATA 01100110
2800 DATA 01111111
2810 DATA 01100111
2820 DATA 01100111
2830 DATA 01111111
2840 DATA 00000000
2850 REM C
2860 DATA 67
2870 DATA 01111111
2880 DATA 01100111
2890 DATA 01100111
2900 DATA 01100000
2910 DATA 01100011
2920 DATA 01100011
2930 DATA 01111111
2940 DATA 00000000
2950 REM D
2960 DATA 68
2970 DATA 01111110
2980 DATA 01100110
2990 DATA 01100110
3000 DATA 01110111
3010 DATA 01110111
3020 DATA 01110111
3030 DATA 01111111
3040 DATA 00000000
3050 REM E
3060 DATA 69
3070 DATA 01111111
3080 DATA 01100000
3090 DATA 01100000
3100 DATA 01111111
3110 DATA 01110000
3120 DATA 01110000
3130 DATA 01111111
3140 DATA 00000000
```

```
3150 REM F
3160 DATA 7Ø
3170 DATA Ø1111111
3180 DATA Ø11ØØØØØ
3190 DATA Ø11ØØØØØ
3200 DATA Ø1111111
3210 DATA Ø111ØØØØ
3220 DATA Ø111ØØØØ
3230 DATA Ø111ØØØØ
3240 DATA ØØØØØØØØ
3250 REM G
3260 DATA 71
3270 DATA Ø1111111
3280 DATA Ø11ØØØ11
3290 DATA Ø11ØØØØØ
3300 DATA Ø11Ø1111
3310 DATA Ø11ØØ111
3320 DATA Ø11ØØ111
3330 DATA Ø1111111
3340 DATA ØØØØØØØØ
3350 REM H
3360 DATA 72
3370 DATA Ø111ØØ11
3380 DATA Ø111ØØ11
3390 DATA Ø111ØØ11
3400 DATA Ø1111111
3410 DATA Ø111ØØ11
3420 DATA Ø111ØØ11
3430 DATA Ø111ØØ11
3440 DATA ØØØØØØØØ
3450 REM I
3460 DATA 73
3470 DATA ØØØØ11ØØ
3480 DATA ØØØØ11ØØ
3490 DATA ØØØØ11ØØ
3500 DATA ØØØØ11ØØ
3510 DATA ØØ1111ØØ
3520 DATA ØØ1111ØØ
3530 DATA ØØ1111ØØ
3540 DATA ØØØØØØØØ
```

```
3550 REM J
3560 DATA 74
3570 DATA 00001100
3580 DATA 00001100
3590 DATA 00001100
3600 DATA 00001110
3610 DATA 00001110
3620 DATA 01101110
3630 DATA 01111111
3640 DATA 00000000
3650 REM K
3660 DATA 75
3670 DATA 01100110
3680 DATA 01100110
3690 DATA 01101100
3700 DATA 01111111
3710 DATA 01100111
3720 DATA 01100111
3730 DATA 01100111
3740 DATA 00000000
3750 REM L
3760 DATA 76
3770 DATA 00110000
3780 DATA 00110000
3790 DATA 00110000
3800 DATA 01110000
3810 DATA 01110000
3820 DATA 01110000
3830 DATA 01111110
3840 DATA 00000000
3850 REM M
3860 DATA 77
3870 DATA 01100111
3880 DATA 01111111
3890 DATA 01111111
3900 DATA 01110111
3910 DATA 01100111
3920 DATA 01100111
3930 DATA 01100111
3940 DATA 00000000
```

```
395Ø REM N
396Ø DATA 78
397Ø DATA Ø11ØØ111
398Ø DATA Ø111Ø111
399Ø DATA Ø1111111
4ØØØ DATA Ø11Ø1111
4Ø1Ø DATA Ø11ØØ111
4Ø2Ø DATA Ø11ØØ111
4Ø3Ø DATA Ø11ØØ111
4Ø4Ø DATA ØØØØØØØØ
4Ø5Ø REM O
4Ø6Ø DATA 79
4Ø7Ø DATA Ø1111111
4Ø8Ø DATA Ø11ØØØ11
4Ø9Ø DATA Ø11ØØØ11
41ØØ DATA Ø11ØØ111
411Ø DATA Ø11ØØ111
412Ø DATA Ø11ØØ111
413Ø DATA Ø1111111
414Ø DATA ØØØØØØØØ
415Ø REM P
416Ø DATA 8Ø
417Ø DATA Ø1111111
418Ø DATA Ø11ØØØ11
419Ø DATA Ø11ØØØ11
42ØØ DATA Ø1111111
421Ø DATA Ø111ØØØØ
422Ø DATA Ø111ØØØØ
423Ø DATA Ø111ØØØØ
424Ø DATA ØØØØØØØØ
425Ø REM Q
426Ø DATA 81
427Ø DATA Ø1111111
428Ø DATA Ø11ØØØ11
429Ø DATA Ø11ØØØ11
43ØØ DATA Ø11ØØ111
431Ø DATA Ø11ØØ111
432Ø DATA Ø11ØØ111
433Ø DATA Ø1111111
434Ø DATA ØØØØØ111
```

```
4350 REM R
4360 DATA 82
4370 DATA 01111110
4380 DATA 01100110
4390 DATA 01100110
4400 DATA 01111111
4410 DATA 01110111
4420 DATA 01110111
4430 DATA 01110111
4440 DATA 00000000
4450 REM S
4460 DATA 83
4470 DATA 01111111
4480 DATA 01100000
4490 DATA 01111111
4500 DATA 00000011
4510 DATA 01110011
4520 DATA 01110011
4530 DATA 01111111
4540 DATA 00000000
4550 REM T
4560 DATA 84
4570 DATA 01111111
4580 DATA 00011100
4590 DATA 00011100
4600 DATA 00011100
4610 DATA 00011100
4620 DATA 00011100
4630 DATA 00011100
4640 DATA 00000000
4650 REM U
4660 DATA 85
4670 DATA 01100111
4680 DATA 01100111
4690 DATA 01100111
4700 DATA 01100111
4710 DATA 01100111
4720 DATA 01100111
4730 DATA 01111111
4740 DATA 00000000
```

```
475Ø REM V
476Ø DATA 86
477Ø DATA Ø11ØØ111
478Ø DATA Ø11ØØ111
479Ø DATA Ø11ØØ111
48ØØ DATA Ø11ØØ111
481Ø DATA Ø11Ø1111
482Ø DATA ØØ11111Ø
483Ø DATA ØØØ111ØØ
484Ø DATA ØØØØØØØØ
485Ø REM W
486Ø DATA 87
487Ø DATA Ø11ØØ111
488Ø DATA Ø11ØØ111
489Ø DATA Ø11ØØ111
49ØØ DATA Ø11Ø1111
491Ø DATA Ø1111111
492Ø DATA Ø1111111
493Ø DATA Ø11ØØ111
494Ø DATA ØØØØØØØØ
495Ø REM X
496Ø DATA 88
497Ø DATA Ø111ØØ11
498Ø DATA Ø111ØØ11
499Ø DATA Ø111ØØ11
5ØØØ DATA ØØ11111Ø
5Ø1Ø DATA Ø11ØØ111
5Ø2Ø DATA Ø11ØØ111
5Ø3Ø DATA Ø11ØØ111
5Ø4Ø DATA ØØØØØØØØ
5Ø5Ø REM Y
5Ø6Ø DATA 89
5Ø7Ø DATA Ø11ØØ111
5Ø8Ø DATA Ø11ØØ111
5Ø9Ø DATA Ø11ØØ111
51ØØ DATA Ø1111111
511Ø DATA ØØØ111ØØ
512Ø DATA ØØØ111ØØ
513Ø DATA ØØØ111ØØ
514Ø DATA ØØØØØØØØ
```

```
5150 REM Z
5160 DATA 90
5170 DATA 01111111
5180 DATA 01100110
5190 DATA 01101100
5200 DATA 00011000
5210 DATA 00110111
5220 DATA 01100111
5230 DATA 01111111
5240 DATA 00000000
5250 REM a
5260 DATA 97
5270 DATA 00000000
5280 DATA 00000000
5290 DATA 00111110
5300 DATA 00000110
5310 DATA 01111110
5320 DATA 01100110
5330 DATA 01111110
5340 DATA 00000000
5350 REM b
5360 DATA 98
5370 DATA 00000000
5380 DATA 01110000
5390 DATA 01110000
5400 DATA 01111110
5410 DATA 01110110
5420 DATA 01110110
5430 DATA 01111110
5440 DATA 00000000
5450 REM c
5460 DATA 99
5470 DATA 00000000
5480 DATA 00000000
5490 DATA 01111100
5500 DATA 01110000
5510 DATA 01110000
5520 DATA 01110000
5530 DATA 01111100
5540 DATA 00000000
```

```
5550 REM d
5560 DATA 100
5570 DATA 00000000
5580 DATA 00000110
5590 DATA 00000110
5600 DATA 01111110
5610 DATA 01101110
5620 DATA 01101110
5630 DATA 01111110
5640 DATA 00000000
5650 REM e
5660 DATA 101
5670 DATA 00000000
5680 DATA 00000000
5690 DATA 01111110
5700 DATA 01100110
5710 DATA 01111110
5720 DATA 01100000
5730 DATA 01111110
5740 DATA 00000000
5750 REM f
5760 DATA 102
5770 DATA 00000000
5780 DATA 00011110
5790 DATA 00011000
5800 DATA 00111110
5810 DATA 00011000
5820 DATA 00011000
5830 DATA 00011000
5840 DATA 00000000
5850 REM g
5860 DATA 103
5870 DATA 00000000
5880 DATA 00000000
5890 DATA 01111110
5900 DATA 01101110
5910 DATA 01101110
5920 DATA 01111110
5930 DATA 00001110
5940 DATA 01111110
```

```
5950 REM h
5960 DATA 104
5970 DATA 00000000
5980 DATA 01110000
5990 DATA 01110000
6000 DATA 01111100
6010 DATA 01110110
6020 DATA 01110110
6030 DATA 01110110
6040 DATA 00000000
6050 REM i
6060 DATA 105
6070 DATA 00000000
6080 DATA 00011000
6090 DATA 00000000
6100 DATA 00011000
6110 DATA 00011000
6120 DATA 00111100
6130 DATA 00111100
6140 DATA 00000000
6150 REM j
6160 DATA 106
6170 DATA 00000000
6180 DATA 00001110
6190 DATA 00000000
6200 DATA 00001110
6210 DATA 00001110
6220 DATA 00001110
6230 DATA 00001110
6240 DATA 00111110
6250 REM k
6260 DATA 107
6270 DATA 00000000
6280 DATA 01100000
6290 DATA 01100000
6300 DATA 01101100
6310 DATA 01111000
6320 DATA 01101100
6330 DATA 01100110
6340 DATA 00000000
```

```
6350 REM l
6360 DATA 108
6370 DATA 00000000
6380 DATA 00111100
6390 DATA 00011100
6400 DATA 00011100
6410 DATA 00011100
6420 DATA 00011100
6430 DATA 00011100
6440 DATA 00000000
6450 REM m
6460 DATA 109
6470 DATA 00000000
6480 DATA 00000000
6490 DATA 01100111
6500 DATA 01111111
6510 DATA 01111111
6520 DATA 01101011
6530 DATA 01100011
6540 DATA 00000000
6550 REM n
6560 DATA 110
6570 DATA 00000000
6580 DATA 00000000
6590 DATA 01111110
6600 DATA 01111110
6610 DATA 01100110
6620 DATA 01100110
6630 DATA 01100110
6640 DATA 00000000
6650 REM o
6660 DATA 111
6670 DATA 00000000
6680 DATA 00000000
6690 DATA 01111110
6700 DATA 01101110
6710 DATA 01101110
6720 DATA 01101110
6730 DATA 01111110
6740 DATA 00000000
```

```
6750 REM p
6760 DATA 112
6770 DATA 00000000
6780 DATA 00000000
6790 DATA 01111110
6800 DATA 01110110
6810 DATA 01110110
6820 DATA 01111110
6830 DATA 01110000
6840 DATA 01110000
6850 REM q
6860 DATA 113
6870 DATA 00000000
6880 DATA 00000000
6890 DATA 01111110
6900 DATA 01101110
6910 DATA 01101110
6920 DATA 01111110
6930 DATA 00001110
6940 DATA 00001110
6950 REM r
6960 DATA 114
6970 DATA 00000000
6980 DATA 00000000
6990 DATA 01111110
7000 DATA 01110110
7010 DATA 01110000
7020 DATA 01110000
7030 DATA 01110000
7040 DATA 00000000
7050 REM s
7060 DATA 115
7070 DATA 00000000
7080 DATA 00000000
7090 DATA 01111110
7100 DATA 01100000
7110 DATA 01111110
7120 DATA 00001110
7130 DATA 01111110
7140 DATA 00000000
```

```
7150 REM t
7160 DATA 116
7170 DATA 00000000
7180 DATA 00111000
7190 DATA 01111110
7200 DATA 00111000
7210 DATA 00111000
7220 DATA 00111000
7230 DATA 00111110
7240 DATA 00000000
7250 REM u
7260 DATA 117
7270 DATA 00000000
7280 DATA 00000000
7290 DATA 01101110
7300 DATA 01101110
7310 DATA 01101110
7320 DATA 01101110
7330 DATA 01111110
7340 DATA 00000000
7350 REM v
7360 DATA 118
7370 DATA 00000000
7380 DATA 00000000
7390 DATA 01101110
7400 DATA 01101110
7410 DATA 01101110
7420 DATA 00111100
7430 DATA 00011000
7440 DATA 00000000
7450 REM w
7460 DATA 119
7470 DATA 00000000
7480 DATA 00000000
7490 DATA 01100011
7500 DATA 01101011
7510 DATA 01111111
7520 DATA 01111111
7530 DATA 00110110
7540 DATA 00000000
```

```
7550 REM x
7560 DATA 120
7570 DATA 00000000
7580 DATA 00000000
7590 DATA 01100110
7600 DATA 00111100
7610 DATA 00011000
7620 DATA 00111100
7630 DATA 01100110
7640 DATA 00000000
7650 REM y
7660 DATA 121
7670 DATA 00000000
7680 DATA 00000000
7690 DATA 01101110
7700 DATA 01101110
7710 DATA 01101110
7720 DATA 01111110
7730 DATA 00001110
7740 DATA 01111110
7750 REM z
7760 DATA 122
7770 DATA 00000000
7780 DATA 00000000
7790 DATA 01111110
7800 DATA 00001100
7810 DATA 00011000
7820 DATA 00110000
7830 DATA 01111110
7840 DATA 00000000
```

Errormeldingen

Vaak lijkt het wel te spoken. Men besteedt veel tijd aan het voorbereiden van zijn BASIC-programma, toetst het zo nauwkeurig mogelijk in de computer in, en toch treden er af en toe na het runnen fouten op.

Omdat uw MSX-computer oorspronkelijk niet uit Nederland stamt, worden de foutmeldingen vanzelfsprekend in het Engels gedaan. Omdat uw MSX-computer bovendien niet over een onbeperkte geheugencapaciteit beschikt, bestaan deze foutmeldingen vaak uit een erg korte aanwijzing. Zoals: 'SYNTAX ERROR' , 'TYPE MISMATCH' e.d.

Deze aanwijzingen leert de ijverige computerprogrammeur mettertijd goed herkennen en begrijpen. Maar voor het zover is, zal er nog heel wat water naar de zee gevloeid zijn. Om deze reden hebben we een uitvoerig errormeldingsprogramma in deze 'verzamelde programma's ' opgenomen. Omdat uw MSX-computer het commando 'ON ERROR GOTO' kent, kan hij bij een optredende fout zo leren reageren als wij het graag zouden zien. Hij kan zelfs voor een deel de fout corrigeren. Een goed voorbeeld hiervan is het volgende. Uw MSX-computer laadt data van een cassette. Uiteindelijk zijn alle data ingelezen. Normaal gesproken zou bij een verdere inleespoging de foutmelding 'INPUT PAST END' gegeven worden. Het is echter veel eenvoudiger vanuit een errorbehandelingsroutine na het optreden van deze fout 'ON ERROR GOTO' het nog geopende datafile af te sluiten en met het programma door te gaan op een punt dat op dat ogenblik het meest logische lijkt.

Ons errormeldingsprogramma kan echter geen fouten verwijderen. Dat zou namelijk al gauw tot onvoorspelbare gevolgen kunnen leiden. Het helpt ons wel op een effectieve manier naar de fouten te zoeken. Het beste kunt u het een keer uitproberen. Type als voorbeeld in regel 20 eens 'NEXT N' in. Als we nu het programma starten, dan wordt meteen de fout (Error 1 = 'NEXT WITHOUT FOR') afgebeeld en ons een mogelijke oplossing aan de hand gedaan.

Hoe werkt het nu met uw reeds bestaande programma's? Allereerst de regels zo hernummeren, commando 'RENUM', dat geen regels het errorbehandelingsprogramma kunnen overlappen. De regelnummers moeten groter dan 20 en kleiner dan 10000 zijn. Hierna gebruikt u het commando 'MERGE', zie tevens uw handleiding, ten einde de beide programma's samen te voegen. Tenslotte met 'RUN' starten en proberen de foutmeldingen met de aangegeven correctieadviezen te gebruiken.

Behalve met het BASIC-commando 'ON ERROR GOTO' kan uw MSX-computer u nog verder helpen door het opvragen van de error-regelvariabele 'ERL' en de foutmeldingsvariabele 'ERR'. De fouten zelf laten zien, kunnen we ten slotte nog met het commando 'ERROR'.

```
fout in regel 20 !
NEXT without FOR

In regel 20 staat het commando
'NEXT', zonder dat voordien in het
programma een bijbehorend FOR-commando
gegeven werd. Het kan ook zijn dat u voor
de bijbehorende FOR -routine een verkeerde
variabele met NEXT geadresseerd heeft.
Ok
*
```

```
10 REM Errormeldingen
20 REM MSX Vezamelde programma's
30 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
40 KEY OFF:SCREEN 0:WIDTH 40:PRINT"Een o
genblikje geduld a.u.b.":GOSUB 11940:COL
OR 15,1:PRINT:PRINT "Het is aan te raden
 regelnummer 40 nu   te verwijderen als
het scherm niet meer veranderd wordt."
50 ON ERROR GOTO 10000
60 REM Het programma tussen de
        regelnummers 70 en 9999 intypen
        of mergen.
9999 STOP
10000 REM Begin van de uitvoerige error
          behandelings-routine
10010 CLS
10020 PRINT"fout in regel"ERL"!"
10030 GOSUB 11570
10040 REM Next without for
10050 IF ERR=1 THEN PRINT "In regel"ERL"
 staat het commando" ELSE GOTO 10100
10060 A$="'NEXT'":A=0:GOSUB 12060:PRINT
", zonder dat voordien in het"
10070 PRINT "programma een bijbehorend '
FOR'-          commando gegeven werd. Het
kan ook zijn dat u voor de bijbehorende
'FOR'-         routine de verkeerde variab
ele met        'NEXT' geadresseerd heeft."
10080 END
10090 REM Syntax error
10100 IF ERR=2 THEN PRINT "Ergens is in
de zinsbouw van regel"ERL ELSE GOTO 1015
0
10110 PRINT "iets verkeerd. b.v. ergens
een woord    fout geschreven. Men spreek
t van een":A$="SYNTAX ERROR":A=0:GOSUB 1
2060:PRINT"."
```

```
10120 PRINT
10130 END
10140 REM Return without gosub
10150 IF ERR=3 THEN PRINT "In regel"ERL"
staat het commando":A$="RETURN":A=0:GOSU
B 12060:PRINT ", zonder dat voordien in"
ELSE GOTO 10190
10160 PRINT "het programma een bijbehore
nde routine  met 'GOSUB' werd aangeroepe
n."
10170 END
10180 REM Out of data
10190 IF ERR=4 THEN PRINT "U wilde nog m
eer ";:A$="DATA":A=0:GOSUB 12060:PRINT "
 inlezen." ELSE GOTO 10230
10200 PRINT "Er zijn echter te weinig da
ta aanwezig. Ook is het mogelijk dat u p
robeerde      met een tweede 'READ' comma
ndo           de zelfde data in te lezen
zonder dat u het commando 'RESTORE' heef
t gebruikt."
10210 END
10220 REM Illegal function call
10230 IF ERR=5 THEN PRINT "Een parameter
 of een ";:A$="'argument'":A=0:GOSUB1206
0:PRINT " werd" ELSE GOTO 10270
10240 PRINT "verkeerd aangegeven. U kunt
 b.v. de      verkeerde waardes voor het
'SOUND' of    'DRAW' commando gebruikt he
bben. Kijk    hiervoor het handboek na!"
10250 END
10260 REM Overflow
10270 IF ERR=6 THEN PRINT "Uw MSX-comput
er registreerde een":A$="OVERFLOW":A=0:G
OSUB 12060:PRINT " dat betekent" ELSE GO
TO 10320
```

```
10280 PRINT "dat in regel"ERL"geprobeerd
 werd een":PRINT"getal samen te stellen
wat te groot is  voor uw computer. Grote
r dan 1.7 E+63.  Ook is het mogelijk dat
 u getallen      probeerde om te zetten
naar een ander   talstelsel en hierbij d
e maximum
10290 PRINT "waarde overschreden heeft."
10300 END
10310 REM Out of memmory
10320 IF ERR=7 THEN PRINT "Het geheugen
ofwel in engels ":A$="MEMORY":A=0:GOSUB
12060:PRINT " is vol." ELSE GOTO 10380
10330 PRINT"Dat is een sterk staaltje u
heeft het      geheugen met een gewoon BASI
C programma volgekregen. Het beste kunt
u het          programma nog eens nalopen.
Om te zien     of het ergens in te korten v
alt.           REM-regels zijn slechts voor
10340 PRINT "de dokumentatie van het pro
gramma          nodig. U kunt overigens ook
 snel het       geheugen vol krijgen als u
variabelen      te groot dimensioneert. Of
te veel         subroutines of 'FOR NEXT' l
ussen           gebruikt.
10350 PRINT "Ook het gebruiken van 'GOSU
B' zonder       dat een 'RETURN' ooit wordt
 uitgevoerd     geeft een dergelijke foutme
lding
10360 END
10370 REM Undefined line number
10380 IF ERR=8 THEN PRINT "U wilt naar e
en programma-regel             springen die
er helemaal niet is.           Ofwel in het
Engels een:":A$="UNDEFINED LINE NUMBER":
A=0:GOSUB 12060:PRINT "." ELSE GOTO 1042
0
10390 PRINT "Misschien heeft u een stap
te ver          gedacht of het programma hi
ervoor          omgenummerd en een gelijkso
ortige fout-    melding over het hoofd gezi
en."
```

```
10400 END
10410 REM Subscript out of range
10420 IF ERR=9 THEN PRINT "Het ";:A$="Su
pscript":A=0:GOSUB 12060:PRINT " (het aa
ntal van de" ELSE GOTO 10470
10430 PRINT"gekozen variabelen) is in re
gel";ERL:PRINT "te groot. Of u heeft ver
geten de varia- belen met 'DIM' te dimen
sioneren of u   dacht dat uw MSX-compute
r meer dan 11
10440 PRINT"variabelen zonder ze te dime
nsioneren   verwerken kan."
10450 END
10460 REM Redemensioned Array
10470 IF ERR=10 THEN PRINT "Het dimensio
neren met ";:A$="DIM":A=0:GOSUB 12060: P
RINT " werd" ELSE GOTO 10510
10480 PRINT"al eens eerder in het progra
mma gedaan. Het is niet mogelijk dat in
regel";ERL: PRINT"nog een keer te doen z
onder voordien    'CLEAR' te hebben gebr
uikt."
10490 END
10500 REM Division by zero
10510 IF ERR=11 THEN A$="DIVISION":A=0:G
OSUB 12060:PRINT " door 0 is niet toeges
taan.":PRINT "Hier is het dan toch gebeu
rd! Kijk in   het programma hoe het geko
men is (ook   de variabelen in de verkee
rde regel     controleren).":ELSE GOTO 1
0540
10520 END
10530 REM Illegal direct
10540 IF ERR=12 THEN PRINT "In de direkt
 mode (";:A$="illegal":A=0:GOSUB 12060:P
RINT ") is dit zo   niet toegestaan." EL
SE GOTO 10580
```

```
10550 PRINT"Dit commando alleen in een p
rogramma    gebruiken en dit met 'RUN' s
tarten."
10560 END
10570 REM Type mismatch
10580 IF ERR=13 THEN:PRINT "U heeft het
verkeerde type variable":PRINT "(";:A$="
TYPE MISMATCH":A=0:GOSUB 12060:PRINT ")
gekozen.":PRINT "Ofwel men vraagt met ee
n variabele       slechts naar een getal
(variabelen       A,B,C...)"ELSE GOTO 106
30
10590 PRINT "en leest hierna slechts een
 getal in     (d.m.v. 'READ' of 'INPUT')
, of men      vewacht alphanummerieke geg
evens b.v     een woord en leest dit in m
et een        stringvariable (A$,B$,C$)."
10600 PRINT "Wil men een getal als strin
g inlezen     dan kan de string door gebr
uik te maken van het commando 'VAL(..)'
weer in een  getal-variabele veranderd w
orden."
10610 END
10620 REM Out of string space
10630 IF ERR=14 THEN A$="STRINGS":A=0:GO
SUB 12060:PRINT " Ik vind geen plaats me
er":PRINT "in het geheugen. Er zijn twee
             mogelijkheden voor verdere st
ring-        invoer:" ELSE GOTO 10680
10640 PRINT:PRINT "1) Met 'CLEAR' meer g
eheugen-ruimte          voor stringsvariab
elen maken.          2) De bestaande strin
gsvariabelen op         cassette saven en
dan met 'CLEAR'         de variabelen verw
ijderen."
10650 PRINT"   Hierna kunnen weer nieuwe
 worden       ingevoerd."
10660 END
```

```
10670 REM String too long
10680 IF ERR=15 THEN PRINT "De in de reg
el";ERL;"gebruikte":PRINT "karakterreeks
 (";:A$="STRING":A=0:GOSUB 12060:PRINT "
) is langer dan"ELSE GOTO 10720
10690 PRINT"255 tekens. Dit moet u vermi
jden!          Daarom strings niet steeds u
itbreiden      (d.m.v. '+') en ook niet te
veel de        stringmanipulatie-functies a
ls 'INSTR'  e.d. gebruiken om strings te
 vergroten."
10700 END
10710 REM String formula too complex
10720 IF ERR=16 THEN PRINT "De karakter-
reeks in regel";ERL:PRINT  "is ";:A$="te
 complex":A=0:GOSUB 12060:PRINT" De stri
ngs niet te           gecompliceerd";ELSE G
OTO 10760
10730 PRINT"maken door het gebruik van s
tringmanipulaties als 'LEFT$', 'RIGHT$',
'MID$' of 'INSTR'.
10740 END
10750 REM Can't continue
10760 IF ERR=17 THEN A$="cont":A=0:GOSUB
 12060:PRINT "inue is welliswaar een han
dige       functie," ELSE GOTO 10810
10770 PRINT "die er voor zorgt dat na 'E
ND', 'STOP'  of een programmaonderbrekin
g door het   toetsen van de <CTRL STOP>-
toetsen       het programma weer gewoon d
oorgaat.     Maar wees er toch maar VOOR
ZICHTIG mee!"
10780 PRINT "In dit geval is er sind de
onderbreking van het programma toch wat
te veel       gebeurd. Zodat de computer
nu niet       meer weet in welke regel hi
j verder      moet."
10790 END
```

```
10800 REM Undefined user function
10810 IF ERR=18 THEN A$="FN":A=0:GOSUB 1
2060:PRINT " is dan wel een mooi command
o om" ELSE GOTO10870
10820 PRINT "rekenkundige functies (die
d.m.v.          'DEF FN' elders in het prog
ramma werden benoemd) met gekozen variab
elen iets uitte rekenen. Maar in dit gev
al werd in    regel";ERL;"met FN een func
tie"
10830 PRINT "aangeroepen die voordien he
lemaal niet  zo werd gedefinieerd.":PRIN
T:PRINT "Daarom een aanwijzing t.b.v. de
 over-    zichtelijkheid:"
10840 PRINT "Het definieren moet altijd
aan het         begin van het programma pla
atsvinden."
10850 END
10860 REM No Resume
10870 IF ERR=21 THEN A$="RESUME":A=0:GOS
UB 12060:PRINT " Dit moet bij een interu
pt"ELSE GOTO 10920
10880 PRINT "routine samen met 'ON ERROR
 GOTO'          gebruikt worden om tijdens
het runnen     van het programma weer op d
e juiste       plaats door te kunnen gaan.
 U kunt        simpel 'RESUME' in het prog
ramma          invoeren hierna gaat het pr
ogramma
10890 PRINT"verder op de plaats waar de
fout           opgetreden is. U kunt ook 'R
ESUME          regelnumer' opgeven om bij e
en gewenste plaats verder te gaan."
10900 END
10910 REM Resume without error
10920 IF ERR=22 THEN PRINT "Het commando
 ";:A$="RESUME":A=0:GOSUB 12060:PRINT "
is in regel"ELSE GOTO 10970
```

```
10930 PRINT ERL;"voorgekomen, zonder dat
 een":PRINT "fout-routine werd ingevoerd
 m.b.v het   commando 'ON ERROR GOTO') o
f in het     programma werd aangeroepen.
 De oorzaak  van de schijnbaar overbodig
e 'RESUME'   kan ook zijn dat het progra
mma niet met"
10940 PRINT "'END' werd afgesloten en hi
erdoor in    de foutbehandelings-routine
 is gestrand."
10950 END
10960 REM Direct statement in file
10970 IF ERR=57 THEN PRINT "Tijdens het
laden van het programma met de cassette
werd een commando gevonden":PRINT"(";:A$
="DIRECTSTATEMENT":A=0:GOSUB 12060:PRINT
 "), wat zonder" ELSE GOTO 11010
10980 PRINT "regelnumer in het programma
 voorkomt.   Dat mag natuurlijk niet!"
10990 END
11000 REM Missing operand
11010 IF ERR=24 THEN PRINT "Veel BASIC-c
ommando's werken niet        zonder dat m
en een minimum aantal ":A$="OPERANDEN":A
=0:GOSUB 12060:PRINT" heeft gegeven. B.v
.het          commando" ELSE GOTO 11030
11020 PRINT"'SOUND' moet minstens gevolg
d worden     door het gewenste kanaal en
de toon-     hoogte.":PRINT "Zoals 'SOUND
 1,100' en niet alleen      'SOUND'.":END
11030 REM Line buffer overflow
11040 IF ERR=25 THEN PRINT "De programar
egel is te lang (op z'n      Engels: ";:A
$="BUFFER OVERFLOW":GOSUB 12060:PRINT ")
; Deze mag" ELSE GOTO 11080
11050 PRINT "maximaal 255 tekens bevatte
n.":PRINT "S.V.P. corrigeren door deze r
egel over  meerdere kleine regels te ver
delen.":END
```

```
11060 END
11070 REM Input past end
11080 IF ERR=55 THEN GOTO 11090 ELSE GOT
O 11160
11090 PRINT "Er werd per ongeluk in rege
l";ERL;"       geprobeerd om datagegevens
in te lezen"
11100 PRINT "langer dan ze in werkelijkh
eid zijn":PRINT "Om deze fout te voorkom
en bestaan de     volgende mogelijkheden:
"
11110 PRINT "Voor het inlezen van data v
anaf de        cassette in een regel vrage
n:             'IF EOF(...) THEN CLOSE:GOT
O ....'"
11120 PRINT "Waneer 'EOF' bereikt wordt,
 wordt de      datareeks netjes afgesloten
 en het        programma gaat op een ander
e plaats       gewoon door. Een andere mog
elijkheid      is het aantal data eerst va
st te stellenen dit getal vooraan weg te
 schrijven"
11130 PRINT "tijdens het saven. Bij het
laden dan      dit getal in een 'FOR...NEX
T' lus         gebruiken. Zodat tevoren be
kend is        hoeveel data ingelezen moet
en worden."
11140 END
11150 REM Bad file name
11160 IF ERR=56 THEN PRINT "De soort v.h
. gekozen file (";:A$="FILE":A=0:GOSUB 1
2060:PRINT:A$="NAME":A=0:GOSUB 12060:PRI
NT ") is niet juist." ELSE GOTO 11200
11170 PRINT "Het inlezen van data met he
t commando    'OPEN...INPUT' is alleen me
t ASCII-data mogelijk. 'CLOAD' laadt all
een maar       programma's die met 'CSAVE'
 werden        gesaved. Hier wordt namelij
k niet lettervoor letter"
```

```
11180 PRINT"weggeschreven maar worden BA
SIC woorden ofwel besturingskarakters we
ggeschreven.En hierdoor afgekort opgesla
gen."
11190 END
11200 REM File already open
11210 IF ERR=54 THEN PRINT "Een file wat
 met het commando ":A$="OPEN":A=0:GOSUB
12060 ELSE GOTO 11260
11220 PRINT" reeds geopend is, mag in":P
RINT "regel";ERL;"niet nog eens met 'OPE
N'"
11230 PRINT"voor INPUT of OUTPUT geopend
 worden.     Dat is bovendien zinloos. Mi
sschien zit de fout in het feit dat de f
ile in een  ander programmadeel of na he
t opnieuw   starten van het programma we
lliswaar"
11240 PRINT "werd geopend maar nog niet
werd           afgesloten met het commando
 'CLOSE'."
11250 END
11260 REM Device i/o error
11270 IF ERR=19 THEN PRINT "U heeft de i
n- uitvoer (Engels:";:A$="I/O":A=0:GOSUB
 12060:PRINT ")":PRINT "van aangesloten
randapparatuur onder-   broken. Dat mag
eigenlijk niet want      de uitgevoerde b
ewerking kan nu niet" ELSE GOTO 11290
11280 PRINT"worden voortgezet. Gevaarlij
k is een     dergelijke onderbreking voor
al bij het  saven van data. Sluit in ied
er geval    nog geopende files af met 'C
LOSE'.":END
11290 REM Verify error
```

```
11300 IF ERR=20 THEN PRINT "U heeft uw M
SX-computer opdracht         gegeven om e
en reeds in het geheugen    aanwezig pro
gramma met een gesaved pro- gramma te ve
rgelijken. (";:A$="VERIFY":A=0:GOSUB 120
60:PRINT ")" ELSE GOTO 11320
11310 PRINT"U heeft voor deze vergelijki
ng het       commando 'CLOAD?' gebruikt.
Nu is er     een 'ERROR' opgetreden. Save
 de          geheugeninhoud nog een keer.
 Misschien   lukt het beter met een ander
e cassette!":END
11320 REM Field overflow
11330 IF ERR=50 THEN PRINT "U heeft bij
het saven op diskette van   een 'relativ
e file' er niet aan gedacht dat het voor
af vastgestelde formaat":PRINT"(";:A$="F
IELD":A=0:GOSUB 12060:PRINT ")" ELSE GOT
O 11360
11340 PRINT "met het data formaat overee
n moet komen."
11350 END
11360 REM Internal error
11370 IF ERR=51 THEN PRINT "Deze fout be
rust op het feit dat binnen in de comput
er iets niet zo gefunctio-  neerd heeft
als eigenlijk zou moeten.   Dit kan u bi
j een onoverwogen gebruik   van het comm
ando 'OUT' heel snel" ELSE GOTO 11400
11380 PRINT"overkomen.
             Een interne fout - Engels:":
A$="INTERNAL ERROR":A=0:GOSUB 12060:PRIN
T " Kijk het in het handboek na."
11390 END
11400 REM Bad file number
```

```
11410 IF ERR=52 THEN PRINT "U mag alleen
 files gebruiken die u via  een 'OPEN' t
oegankelijk heeft gemaakt.  Dus eerst 'O
PEN...FOR INPUT/OUTPUT' en  als afsluiti
ng van de werkzaamheden     'CLOSE' niet
 vergeten." ELSE GOTO 11440
11420 PRINT"Bovendien moet u er op lette
n dat u      niet meer files kunt openen
dan u voor- heen met 'MAXIFILES' benoemd
 heeft."
11430 END
11440 REM File not found
11450 IF ERR=53 THEN PRINT "Deze 'ERROR'
 treedt alleen bij het       werken met d
iskette's op. U wilt een    file of prog
ramma in de computer laden  wat op de ge
plaatste diskette helmaal   niet voorkom
t."ELSE GOTO 11480
11460 PRINT "Vraagt u de inhoudsopgave v
an de disk   maar eens op met 'FILES'. O
f als u met  twee diskdrives werkt met '
FILES2'."
11470 END
11480 REM Sequential i/o only
11490 IF ERR=58 THENPRINT "U moet onders
cheid maken tussen" :A$="sequentials":A=
0:GOSUB 12060 :PRINT:PRINT "en relative
files. Bij een sequential   worden alle
data achter elkaar          weggeschreve
n."ELSE GOTO 11520
11500 PRINT "Bij een relative file zijn
de data       direct adresseerbaar en heb
ben een       vooraf vastgelegd formaat!"
11510 END
11520 REM File not open
11530 IF ERR=59 THEN PRINT "U moet een f
ile eerst met het commando  ";:A$="OPEN"
:A=0:GOSUB 12060:PRINT " openen voordat
u ze met" ELSE GOTO 11560
```

```
11540 PRINT "de commando's zoals 'PRINT
#' en          'INPUT #' gaat gebruiken."
11550 END
11560 PRINT "Onbekende foutmelding":END
11570 REM De errormeldingen met hun Enge
lstalge korte omschrijving
11580 IF ERR=1 THEN PRINT"NEXT without F
OR"
11590 IF ERR=2 THEN PRINT"Syntax error"
11600 IF ERR=3 THEN PRINT"RETURN without
 GOSUB"
11610 IF ERR=4 THEN PRINT"OUT of DATA"
11620 IF ERR=5 THEN PRINT"Illegal functi
on call"
11630 IF ERR=6 THEN PRINT"Overflow"
11640 IF ERR=7 THEN PRINT"Out of memory"
11650 IF ERR=8 THEN PRINT"Undefined line
number"
11660 IF ERR=9 THEN PRINT"Subscript out
of range"
11670 IF ERR=10 THEN PRINT"Redimensioned
 array"
11680 IF ERR=11 THEN PRINT"Division by z
ero"
11690 IF ERR=12 THEN PRINT"Illegal direc
t"
11700 IF ERR=13 THEN PRINT"Type mismatch
"
11710 IF ERR=14 THEN PRINT"Out of string
 space"
11720 IF ERR=15 THEN PRINT"String too lo
ng"
11730 IF ERR=16 THEN PRINT"String formul
a too complex"
11740 IF ERR=17 THEN PRINT"Can't continu
e"
11750 IF ERR=18 THEN PRINT"Undefined use
r function"
```

```
11760 IF ERR=19 THEN PRINT"Device I/O er
ror"
11770 IF ERR=20 THEN PRINT"Verify error"
11780 IF ERR=21 THEN PRINT"No RESUME"
11790 IF ERR=22 THEN PRINT"RESUME withou
t error"
11800 IF ERR=24 THEN PRINT"Missing opera
nd"
11810 IF ERR=25 THEN PRINT"Line buffer o
verflow"
11820 IF ERR=50 THEN PRINT"Field overflo
w"
11830 IF ERR=51 THEN PRINT"Internal erro
r"
11840 IF ERR=52 THEN PRINT"Bad file numb
er"
11850 IF ERR=53 THEN PRINT"File not foun
d"
11860 IF ERR=54 THEN PRINT"File already
open"
11870 IF ERR=55 THEN PRINT"Input past en
d"
11880 IF ERR=56 THEN PRINT"Bad file name
"
11890 IF ERR=57 THEN PRINT"Direct statem
ent in file"
11900 IF ERR=58 THEN PRINT"Sequential I/
O onley"
11910 IF ERR=59 THEN PRINT"File not OPEN
"
11920 PRINT:PRINT
11930 RETURN
11940 REM Omzetten >chr$(127)->revers/ne
gatief
11950 FOR M=2304 TO 3055
11960 A$=BIN$(VPEEK(M))
11970 A$=STRING$(8-LEN(A$),"0")+A$
11980 FOR N=1 TO 8
```

```
11990 IF MID$(A$,N,1)="1" THEN MID$(A$,N
,1)="0" ELSE MID$(A$,N,1)="1"
12000 NEXT N
12010 A$="&b"+A$
12020 VPOKE M+768,VAL(A$)
12030 A$=""
12040 NEXT M
12050 RETURN
12060 REM Veranderen in negatief
          letterbeeld
12070 REM A=0 -> alles
12080 REM A=1 -> tot aan de spatie
12090 REM A=2 -> alleen kleine letters
12100 REM A=3 -> alleen hoofdletters
12110 REM A=4 -> alleen getallen
12120 FOR PQ=1 TO LEN(A$)
12130 IF A=0 THEN GOTO 12180
12140 IF A=1 AND MID$(A$,PQ,Q)=" "THEN G
OTO 12190
12150 IF A=2 AND (MID$(A$,PQ,1)<CHR$(97)
 OR MID$(A$,PQ,1)>CHR$(125)) THEN GOTO 1
2190AX ERROR
12160 IF A=3 AND (MID$(A$,PQ,1)<CHR$(65)
 OR MID$(A$,PQ,1)>CHR$(93)) THEN GOTO 12
190
12170 IF A=4 AND (MID$(A$,PQ,1)>CHR$(57)
 OR MID$(A$,PQ,1)<CHR$(48)) THEN GOTO 12
190
12180 MID$(A$,PQ,1)=CHR$(ASC(MID$(A$,PQ,
1))+96)
12190 NEXT PQ
12200 PRINT A$;
12210 RETURN
```

Variabelenreferentielijst

Bij het programma 'geheugeninhoud 4' hebben we al geleerd hoe het betreffende regelnummer en de geheugenlocatie van de volgende regel in het geheugen wordt opgeslagen. Geheugenplaatsen 32769 en 32770 verwijzen naar de geheugenplaats van de tweede programmaregel. Geheugenplaatsen 32771 en 32772 vertellen ons wat het eerste regelnummer is, enz. Met het programma 'geheugeninhoud 4' bezitten we bovendien een eenvoudige methode het geheugen te laten doorzoeken naar een bepaald woord met een maximale lengte van zes letters. En toch zullen velen van u zich misschien afvragen: 'Wat is nu eigenlijk het praktische nut van dit alles?' Wel, we hebben hierdoor allereerst het geheugen wat beter leren kennen. En van de andere kant kan het programma 'geheugenlocatie 4' ons altijd nog van pas komen, wanneer we naar een dergelijk alfanumeriek gegeven moeten zoeken.

Bij het programma 'variabelenreferentielijst' gaan we met een soortgelijk oogmerk van start, maar deze keer met een aanzienlijke uitbreiding. Ditmaal wordt in het geheugen niet alleen naar een bepaalde karaktervolgorde gezocht. Nee, er wordt bovendien een rangorde samengesteld.

De karaktervolgorde waarnaar gezocht wordt, gaat uit van de aanwezigheid van hoofdletters en cijfers die onder bepaalde omstandigheden alleen maar een variabele kunnen zijn en niets anders.

Voorwaarde 1: Als er in een regel geen aanhalingsteken voorkomt, dan moet het i.g.v. hoofdletters een variabele betreffen.
Voorwaarde 2: Het moet geen REM-regel zijn.
Voorwaarde 3: Voorafgaande aan de hoofdletter staat geen '255', want dan moet het een besturingskarakter zijn.

Omdat uw MSX-computer maar twee karakters per variabele kent, maar ook de mogelijkheid heeft langere namen van variabelen te accepteren, moeten we uiteraard alle variabelen in het geheugen opzoeken. En omdat er verschillende variabelen kunnen voorkomen, zoals numerieke en stringvariabelen, moet ook een eventueel karakter als '!' en '$' meegelezen worden en als uitkomst afgebeeld.

Wat zijn nu de mogelijkheden van ons variabelenreferentielijst-programma?

1) Er wordt naar variabelen in het programmageheugen gezocht. Uw te onderzoeken programma mag als hoogste regelnummer regel 9999 hebben. Het programma 'variabelenreferentielijst' wordt gestart met 'RUN' 10000.

2) Staat de gevonden variabele eenmaal alfabetisch op de eerste plaats, van alle in het programma voorkomende variabelen, dan wordt in het geheugen = programma verder gezocht in welke regelnummers deze zelfde naam ook nog voorkomt. Ook dit wordt op het beeldscherm getoond. Zo zal na enige tijd een alfabetische variabelenlijst op het scherm worden afgebeeld, die ons bij ons 'BASIC-programm-debugging' ook wel foutzoeken of ontvlooien genoemd, aanzienlijk op weg kan helpen.

Probeert u het programma meteen maar eens uit. Voer hiertoe de volgende regels met variabelen in:

```
10 a$="MSX Computer"
20 b$="DATA BECKER"
30 MS=64
40 BE!=100
50 BA%=200
60 D#=4000
```

Nu start u het 'variabelenreferentielijst-programma' met 'RUN 10000' en na een korte tijd worden niet alleen de gebruikte variabelen en hun aantal maar ook het aantal keren dat ze voorkomen en de bijbehorende regelnummers op het beeldscherm afgebeeld.

Is het programma doorzocht, dan geeft uw MSX-computer de gebruikte geheugenruimte op het beeldscherm aan, dat wil zeggen de gebruikte ruimte van het programma zonder het 'variabelenreferentielijst-programma'. Evenals het aantal en de inhoud van de verschillende variabelen.

Nog een paar tips

Als het programma begint, ziet u achtereenvolgens een deel van de karaktertekens veranderen in negatieve karakters. Deze langdurige procedure hoeft u maar één keer over u heen te laten gaan, tenzij u tussentijds het scherm veranderen wilt. Als u het 'SCREEN'-commando niet gebruikt, kunt u de eerste regels overeenkomstig de aanwijzing weer verwijderen, nadat u het programma de eerste maal gestart heeft.

Hoe langer uw programma is, des te uitgebreider en daardoor langduriger moet uw MSX-computer het geheugen doorzoeken. U moet dus een beetje geduld oefenen.

De namen van de variabelen worden eerst alfabetisch gesorteerd. Dan worden de bijbehorende regelnummers toegevoegd, ook als het om meerdere regelnummers gaat. En ten slotte wordt alles in de vorm van een referentielijst op het beeldscherm afgebeeld.

De aangegeven programmalengte heeft alleen betrekking op het programma in de regels kleiner dan 10000. Want de 'referentielijst' hoort tenslotte niet bij het eigenlijke hoofdprogramma. Ze fungeert hier alleen maar als een zgn. utility.

```
  A$  10
----------------------------------------
  B$  20
----------------------------------------
  BA%  50
----------------------------------------
  BE!  40
----------------------------------------
  D≠  60
----------------------------------------
  MS  30
----------------------------------------
========================================
Variabelen totaal: 6
Verschillende variabelen: 6
Lengte v.h. prg. t/m regel 9999: 86 bytes
list 10-60
10 A$="MSX-Computer"
20 B$="DATA BECKER"
30 MS=64
40 BE!=100
50 BA%=200
60 D≠=400
Ok
*
```

```
10000 REM Variabelen referentielijst
10010 REM MSX Vezamelde programma's
10020 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
10030 REM Voorbereidingen treffen
10040 SCREEN 0
10050 WIDTH 40
10060 COLOR 15,1
10070 KEY OFF
10080 REM Het inladen van de negatieve
          karakterset
10090 PRINT "Een ogenblikje a.u.b."
10100 FOR N=32 TO 255
10110 PRINT CHR$(N);
10120 NEXTN
10130 GOSUB 10640
10140 PRINT
10150 PRINT "Als u het scherm niet veran
dert kunt u  de regels 10040 t/m 10160 v
erwijderen."
10160 STOP
10170 REM Het begin van het eigenlijke
          ===========================
          hoofdprogramma
          ==============
10180 CLS
10190 DEFINT A-Z
10200 CLEAR 5000
10210 REM Het dimensioneren van 1000
          variabelen-namen en de bij-
          behorende regelnummers
10220 DIM A$(1000),A(1000)
10230 REM BASIC-programmastart
10240 Z=32769!
10250 V=1
10260 REM Verwijzing naar de volgende
          programma regel in het
          geheugen
10270 A=PEEK(Z)+256*PEEK(Z+1)
10280 REM Regelnummers
```

```
10290 B=PEEK(Z+2)+256*PEEK(Z+3)
10300 REM Er wordt alleen in het
      geheugen t/m regelnummer 9999
      naar variabelen gezocht
10310 IF B=10000 THEN Q1=Z-32769!:GOSUB
10490:GOSUB 10920:PRINT "Variabelen tota
al:";V-1:PRINT "Verschillende variabelen
;";QQ:PRINT "Lengte v.h. prg. t/m regel
9999:";Q1;"bytes":END
10320 FOR N=Z+4 TO A-1
10330 REM kriteria voor het zoeken
          ========================
          in geheugen variabelen j/n
          ===========================
10340 REM 1) tussen 65 en 90 ->
             hoofdleters
          2) op voorgaande geheugen-
             plaats geen 255 ->
             besturingskarakter
10350 REM 3) niet de inhoud van een
             print-regel dus geen "=
             chr$(34) aanwezig
             typering variabele 'ME'
10360 REM 4) Geen remark ofwel REM-regel
             (REM of ')
10370 IF PEEK(N)<91 AND PEEK(N)>64 AND P
EEK(N-1)<>255 AND ME<>1 AND M1<>1 THEN G
OTO 10460
10380 IF PEEK(N)=34 THEN IF ME=0 THEN ME
=1 ELSE ME=0
10390 IF PEEK(N)=230 OR PEEK(N)=143 THEN
 M1=1
10400 NEXT N
10410 ME=0
10420 M1=0
10430 Z=A
10440 GOTO 10270
10450 REM Het samenvoegen van de
          variabelen namen in a$
10460 A$=CHR$(PEEK(N))
```

```
10470 IF (PEEK(N+1)<91 AND PEEK(N+1)>64)
 OR (PEEK(N+1)=33) OR (PEEK(N+1)>34 AND
PEEK(N+1)<38) OR (PEEK(N+1)<58 AND PEEK(
N+1)>47) OR (PEEK(N+1)=40) THEN A$=A$+CH
R$(PEEK(N+1)):N=N+1:GOTO 10470 ELSE A(V)
=B:A$(V)=A$:V=V+1:GOTO 10400
10480 REM Het ordenen van de var. namen
          ==============================
10490 A$="z"
10500 FOR M=1 TO V-1
10510 IF A$(M)<A$ THEN A$=A$(M):A=M
10520 NEXT M
10530 A$(A)="z"
10540 IF A$="z" THEN PRINTSTRING$(40,"="
);:RETURN ELSE Q$=" "+A$+" ":GOSUB 10760
:PRINT A(A);
10550 REM Het toewijzen van een of
          ========================
          meerdere regelnummers waar
          ==========================
          de variabele voorkomt
          =====================
10560 FOR O=1 TO V-1
10570 IF A$(O)=A$ THEN PRINT A(O);:A$(O)
="z"
10580 NEXT O
10590 REM Het beeindigen van de weer-
          gave van een variabele op
          het beeldscherm
10600 PRINT
10610 PRINT STRING$(40,"-");
10620 QQ=QQ+1
10630 GOTO 10490
10640 REM Het veranderen van >chr$(127)
          in negatief
10650 FOR M=2304 TO 2775
10660 A$=BIN$(VPEEK(M))
10670 A$=STRING$(8-LEN(A$),"0")+A$
10680 FOR N=1 TO 8
```

```
10690 IF MID$(A$,N,1)="1" THEN MID$(A$,N
,1)="0" ELSE MID$(A$,N,1)="1"
10700 NEXT N
10710 A$="&b"+A$
10720 VPOKE M+768,VAL(A$)
10730 A$=""
10740 NEXT M
10750 RETURN
10760 REM veranderen in negatiefschrift
10770 REM k=0 -> alles
10780 REM k=1 -> tot aan spatie
10790 REM k=2 -> alleen kleine letters
10800 REM k=3 -> alleen hoofdletters
10810 REM k=4 -> alleen cijfers
10820 FOR PQ=1 TO LEN(Q$)
10830 IF K=0 THEN GOTO 10880
10840 IF K=1 AND MID$(Q$,PQ,1)=" " THEN
GOTO 10890
10850 IF K=2 AND (MID$(Q$,PQ,1)<CHR$(97)
 OR MID$(Q$,PQ,1)>CHR$(125)) THEN GOTO 1
0890
10860 IF K=3 AND (MID$(Q$,PQ,1)<CHR$(65)
 OR MID$(Q$,PQ,1)>CHR$(93)) THEN GOTO 10
890
10870 IF K=4 AND (MID$(Q$,PQ,1)>CHR$(57)
 OR MID$(Q$,PQ,1)<CHR$(48)) THEN GOTO 10
890
10880 MID$(Q$,PQ,1)=CHR$(ASC(MID$(Q$,PQ,
1))+96)
10890 NEXT PQ
10900 PRINT Q$;
10910 RETURN
10920 IF V-1=0 THEN QQ=0
10930 RETURN
```

Kaartenbak voor grammofoonplaten

Dit programma kan zijn naam alleen dan waarmaken wanneer u het woord voor woord en commando voor commando precies zo intypt als het in deze 'verzamelde programma's' staat afgebeeld.

Omdat het eindprodukt een niet onverdienstelijke kleine databank is geworden, is het ook mogelijk de PRINT-commando's, zoals 'PRINT "Platen invoeren"', te veranderen in een door u zelf gekozen tekst bijvoorbeeld: 'PRINT "Adressen invoeren".

Zo beschikken we na een paar veranderingen in het programma over een databank met adressen, die we natuurlijk apart kunnen en moeten saven.

Het programma 'kaartenbak voor grammofoonplaten' heeft echter een aantal beperkingen die we eigenlijk alleen maar behoorlijk kunnen omzeilen als we niet met een cassette maar met een diskette gaan werken. Onze bestanden zijn dan direct toegankelijk.

De grenzen van dit programma laten zich het beste waarnemen als we de vrij beschikbare geheugencapaciteit eens nader bekijken. Willen we een enkele kaart volledig benutten, met 3 maal 40 karakters, dan hebben we daar alleen al 120 geheugenposities voor nodig. Bij honderd kaarten zijn dat al 12000 geheugenplaatsen. En hieruit kunnen we concluderen dat we 300 kaarten als het maximum moeten beschouwen.

Wilt u per kaart minder dan 120 karakters vastleggen dan kunt u het programma uiteraard hieraan aanpassen. In dit geval kunt u de bovengrens eenvoudigweg verhogen. Zie regel 100.

'IF Z < (1 OR Z) > 400 . . . enz.'

Het programma 'kaartenbak voor grammofoonplaten' biedt u de volgende keuzemogelijkheden:

1. U kunt kaarten samenstellen met elk drie velden. (Titel van de plaat, naam van de zanger en het nummer)
2. U kunt de data van cassette laden en
3. U kunt data naar cassette saven.

4. U kunt naar een plaat in het geheugen laten zoeken. Naar titel, zanger of nummer. Hierbij wordt het beeldscherm met de gevonden (data)gegevens gevuld totdat het vol is. Als u hierna op de (ENTER)-toets drukt, volgen de volgende gegevens. Het meerdere malen vinden van gegevens is mogelijk als u bijvoorbeeld meer dan één plaat van dezelfde zanger bezit.

5. U kunt uw hele grammofoonplatenlijst op het beeldscherm, of indien u hierover de beschikking heeft, op een printer laten weergeven. Per beeldscherm verkrijgt u altijd slechts vijf kaarten tegelijkertijd. Hierna op (ENTER) drukken.

6. U kunt reeds ingevoerde data veranderen. Omdat bij keuzemogelijkheid (1.) reeds de mogelijkheid is ingebouwd achteraf correcties aan te brengen ('correct?'), zal deze functie nauwelijks gebruikt hoeven worden om achteraf een foutieve invoer te herstellen. Bij deze functie is er meer van uitgegaan dat u het kaartenbakprogramma, zoals reeds eerder vermeld, gebruikt om adressenbestanden op te slaan. Hoe vaak komt het niet voor dat iemand verhuist? De oplossing is eenvoudig. Adreskaartenbak inladen. Punt 6 = veranderen kiezen. Omdat zeker niet alle gegevens van uw kennissen van vandaag op morgen veranderd zullen zijn, in ieder geval de naam niet, kunt u dit programmadeel door het drukken van de (ENTER)-toets eenvoudig overslaan. Alleen op het punt waar ook daadwerkelijk een verandering is opgetreden, kunt u deze invoeren.

Of u dit programma, zoals oorspronkelijk de bedoeling was, voor het vastleggen van uw grammofoonplaten gebruikt, of dat u het programma ombouwt tot een adressenbestand, we wensen u in ieder geval veel plezier toe met het ten strijde trekken tegen de wanorde met behulp van uw MSX-computer en ons programma.

```
Kaartenbak voor grammofoonplaten

              Menu

  1. Platen invoeren
  2. Platen inladen
  3. Platen saven
  4. Platen zoeken
  5. Platen afbeelden
  6. Platen veranderen

Uw keuze (1/2/3/4/5/6) ?
```

```
4. Platen zoeken

Terug naar het menu d.m.v. ENTER
zonder invoer.

Waar wilt u naar zoeken?

1) Titel v.d. plaat
2) Zanger/es, groep
3) Nummer/code

Uw keuze (1/2/3) ? 1
```

```
10 REM Kaartenbak voor grammofoonplaten
20 REM MSX Verzamelde programma's
30 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
40 SCREEN 0:COLOR 15,1,15:KEY OFF:CLEAR
15000
50 PRINT "Een ogenblikje a.u.b.":GOSUB 2
370:CLS:M=1:WIDTH 40
60 REM Voorbereidingen treffen voor het
       aanmaken van gedimensioneerde
       variabelen
70 A$="Kaartenbak voor grammofoonplaten"
:A=0:GOSUB 2490
80 PRINT
90 INPUT "Hoeveel LP's moeten opgeslagen
 worden   (maximaal 300) ";Z
100 IF Z<1 OR Z>300 THEN GOSUB 2290:GOTO
 50
110 FR=FRE(0):DIM P$(Z),S$(Z),N$(Z)
120 PRINT:PRINT "Alleen al het dimension
eren van";Z:PRINT "platen verbruikt";FR-
FRE(0);"geheugenplaatsen."
130 PRINT:PRINT:PRINT "Er blijven per pl
aat in het gunstigste  geval":PRINT:PRIN
T "          ";INT(17000/Z):PRINT:PRINT
"geheugenplaatsen over.":PRINT:PRINT:PRI
NT
140 INPUT "Is dat voldoende? Anders moet
 u het     aantal van de in te voeren pl
aten       veranderen ( /N) ";F$
150 IF LEFT$(F$,1)="n" OR LEFT$(F$,1)="N
" THEN F$="N"
160 IF F$="N" THEN CLEAR: GOTO 50
170 REM Weergeven van het hoofdmenu
180 CLS
190 LOCATE 5,7
200 A$="Kaartenbak voor grammofoonplaten
":A=0:GOSUB 2490
210 LOCATE 15,9
220 PRINT "Menu"
```

```
230 LOCATE 7,12
240 PRINT "1. Platen invoeren"
250 LOCATE 7,13:PRINT "2. Platen inladen
"
260 LOCATE 7,14:PRINT "3. Platen saven"
270 LOCATE 7,15:PRINT "4. Platen zoeken"
280 LOCATE 7,16:PRINT "5. Platen afbeeld
en"
290 LOCATE 7,17:PRINT "6. Platen verande
ren"
300 LOCATE 5,20
310 INPUT "Uw keuze (1/2/3/4/5/6) ";F$
320 IF F$<"1" OR F$>"6" THEN GOSUB 2290:
GOTO 180
330 ON VAL(F$) GOSUB 370,680,930,1210,17
40,2010
340 F$="":F1$="":F2$="":N1=0
350 GOTO 180
360 REM Routine voor het invoeren
        van data
370 IF M<1 THEN M=1
380 FOR I=M TO Z
390 CLS:A$="1. Platen invoeren":A=0:GOSU
B 2490
400 PRINT
410 GOSUB 2350
420 PRINT
430 F1$="":PRINT "Titel v.d. plaat"
440 INPUT P$(I):IF LEN(P$(I))>40 THEN P$
(I)=LEFT$(P$(I),40)
450 IF P$(I)="" THEN I=I-1: M=I:RETURN
460 PRINT "Zanger/es, groep"
470 INPUT S$(I):IF LEN(S$(I))>40 THEN S$
(I)=LEFT$(S$(I),40)
480 PRINT "Nummer/code"
490 INPUT N$(I):IF LEN(N$(I))>40 THEN N$
(I)=LEFT$(N$(I),40)
500 REM Eerste weergave van de
        ingevoerde data met
        correctie-mogelijkheid
```

```
510 PRINT
520 PRINT "Nogmaals de ingevoerde data:"
530 PRINT "Bezet t/m No.";I;"Vrij t/m NØ
.";Z
540 PRINT
550 A$="Titel v.d. plaat":A=Ø:GOSUB 249Ø
560 PRINT P$(I)
570 A$="Zanger/es, groep:":A=Ø:GOSUB 249
Ø
580 PRINT S$(I)
590 A$="Nummer/code":A=Ø:GOSUB 249Ø
600 PRINT N$(I)
610 PRINT
620 INPUT "Correct ( /n) ";F1$
630 IF LEFT$(F1$,1)="N" OR LEFT$(F1$,1)=
"n" THEN F1$="N"
640 IF F1$="N" THEN CLS:GOTO 43Ø
650 NEXT I
660 RETURN
670 REM Routine voor het laden van
        data van de cassette
680 CLS
690 A$="Platen inladen":A=Ø:GOSUB 249Ø
700 PRINT
710 GOSUB 235Ø
720 PRINT
730 PRINT "Met het inladen wist u alle r
eeds         aanwezige gegevens uit het ge
heugen!"
740 PRINT
750 INPUT "Wilt u gegevens inladen (j/ )
" ;F1$
760 IF LEFT$(F1$,1)="j" OR LEFT$(F1$,1)=
"J" THEN F1$="J"
770 IF F1$="J" THEN GOTO 78Ø ELSE RETURN
780 PRINT
790 PRINT "Bestandsnaam"
800 INPUT DN$
810 OPEN DN$ FOR INPUT AS#1
820 INPUT#1,M
```

```
830 PRINT
840 PRINT "Het in te laden bestand ";DN$
850 PRINT "bestaat uit";M-1;"records
860 FOR N=1 TO M
870 INPUT#1,P$(N),S$(N),N$(N)
880 PRINT N:PRINT P$(N):PRINT S$(N):PRIN
T N$(N)
890 NEXT N
900 CLOSE #1
910 RETURN
920 REM Routine voor het saven van
        data naar de cassette
930 CLS
940 A$="3. Platen saven":A=0:GOSUB 2490
950 PRINT
960 GOSUB 2350
970 PRINT
980 INPUT "Wilt u de gegevens saven (J/
) ";F1$
990 IF LEFT$(F1$,1)="j" OR LEFT$(F1$,1)=
"J" THEN F1$="J"
1000 IF F1$="J" THEN GOTO 1010 ELSE RETU
RN
1010 PRINT
1020 IF DN$<>"" THEN PRINT "Moet onder d
e bestandsnaam ";DN$:INPUT "gesaved word
en (J/ ) ";F2$:IF LEFT$(F2$,1)="j" OR LE
FT$(F2$,1)="J" THEN F2$="J"
1030 IF F2$="J" THEN GOTO 1060
1040 PRINT "Bestandsnaam ";
1050 INPUT DN$
1060 OPEN DN$ FOR OUTPUT AS#1
1070 PRINT #1,M
1080 PRINT
1090 PRINT "Het te saven bestand ";DN$:P
RINT "bestaat uit";M-1;"records"
1100 FOR N=1 TO M
1110 PRINT #1,P$(N):PRINT #1,S$(N):PRINT
 #1,N$(N)
1120 PRINT N
```

```
1130 PRINT P$(N)
1140 PRINT S$(N)
1150 PRINT N$(N)
1160 PRINT
1170 NEXT N
1180 CLOSE #1
1190 RETURN
1200 REM Routine voor het zoeken naar
         Platen, zanger of nummer
1210 CLS
1220 A$="4. Platen zoeken":A=0:GOSUB 249
0
1230 PRINT
1240 GOSUB 2350
1250 PRINT
1260 REM Afbeelden van een hulpmenu
         Zoeken naar:
         1) Naam v.d. plaat
         2) Zanger/es, groep
         3) Numer/code
1270 PRINT "Waar wilt u naar zoeken"
1280 PRINT
1290 PRINT "1. Titel v.d. plaat"
1300 PRINT "2. Zanger/es, groep"
1310 PRINT "3. Numer/code"
1320 PRINT
1330 INPUT "Uw keuze (1/2/3) ";F1$
1340 IF F1$<"1" OR F1$>"3" THEN RETURN
1350 ON VAL(F1$) GOSUB 1380,1500,1610
1360 F1$="":F2$="":N1=0:GOTO 1210
1370 REM Het zoeken naar de titel
         v.d. plaat
1380 CLS
1390 PRINT "Titel v.d. plaat"
1400 INPUT P$:IF LEN(P$)>40 THEN P$=LEFT
$(P$,40)
1410 IF P$="" THEN RETURN
1420 FOR N=1 TO M
1430 IF P$=P$(N) THEN PRINT N:PRINT P$(N
):PRINT S$(N);PRINT N$(N):N=N+1
```

```
1440 IF N1=5 THEN N1=0:GOSUB 2320
1450 NEXT N
1460 PRINT
1470 GOSUB 2320
1480 RETURN
1490 REM Het zoeken naar Zanger/es,
         groep
1500 CLS
1510 PRINT "Zanger/es, groep"
1520 INPUT S$:IF LEN(S$)>40 THEN S$=LEFT
$(S$,40)
1530 IF S$="" THEN RETURN
1540 FOR N=1 TO M
1550 IF S$=S$(N) THEN PRINT N:PRINT P$(N
):PRINT S$(N):PRINT N$(N):N1=N1+1
1560 IF N1=5 THEN N1=0:GOSUB 2320
1570 NEXT N
1580 PRINT
1590 GOSUB 2320
1600 RETURN
1610 CLS
1620 REM Het zoeken naar nummer/code
1630 PRINT "Nummer/code"
1640 INPUT N$
1650 IF N$="" THEN RETURN
1660 FOR N=1 TO M
1670 IF N$=N$(N) THEN PRINT N:PRINT P$(N
):PRINT S$(N):PRINT N$(N):N1=N1+1
1680 IF N1=5 THEN N1=0:GOSUB 2320
1690 NEXT N
1700 PRINT
1710 GOSUB 2320
1720 RETURN
1730 REM Routine voor afbeelden van
         gegevens op beeldscherm of
         printer
```

```
1740 CLS
1750 A$="5. Platen afbeelden":A=0:GOSUB
2490
1760 PRINT
1770 GOSUB 2350
1780 PRINT
1790 PRINT "Op het beeldscherm of op de
printer      afbeelden?"
1800 INPUT "(B/P) ";F2$
1810 IF F2$="" THEN RETURN
1820 IF LEFT$(F2$,1)="b" OR LEFT$(F2$,1)
="B" THEN F2$="B" ELSE IF LEFT$(F2$,1)="
p" OR LEFT$(F2$,1)="P" THEN F2$="P"
1830 IF F2$="P" THEN GOTO 1940
1840 REM Afbeelden op het beeldscherm
1850 CLS:FOR N=1 TO M
1860 N1=N1+1
1870 IF N1=5 THEN N1=0:GOSUB 2320:CLS
1880 PRINT N:PRINT P$(N):PRINT S$(N):PRI
NT N$(N)
1890 NEXT N
1900 PRINT
1910 GOSUB 2320
1920 RETURN
1930 REM Afbeelden op de printer
1940 FOR N=1 TO M-1
1950 LPRINT N:LPRINT P$(N):LPRINT S$(N):
LPRINT N$(N):LPRINT
1960 NEXT N
1970 PRINT
1980 GOSUB 2320
1990 RETURN
2000 REM Routine voor het veranderen
         van gegevens in het geheugen
2010 CLS
2020 A$="6. Platen veranderen":A=0:GOSUB
 2490
```

```
2030 PRINT
2040 GOSUB 2350
2050 PRINT
2060 INPUT "Welk numer ";N1
2070 IF N1>M OR N1<1 THEN GOSUB 2290:GOT
O 2060
2080 IF N1=0 THEN RETURN
2090 PRINT
2100 PRINT N1:PRINT P$(N):PRINT S$(N):PR
INT N$(N)
2110 PRINT
2120 PRINT "Iets veranderen? (Nee=<ENTER
>)"
2130 PRINT P$(N1)
2140 INPUT D$:IF D$<>"" THEN P$(N1)=D$
2150 PRINT S$(N1)
2160 INPUT D$:IF D$<>"" THEN S$(N1)=D$
2170 PRINT N$(N1)
2180 INPUT D$:IF D$<>"" THEN N$(N1)=D$
2190 CLS
2200 PRINT "De invoer voor plaat nummer"
;N1;":"
2210 PRINT P$(N1)
2220 PRINT S$(N1)
2230 PRINT N$(N1)
2240 PRINT
2250 GOSUB 2320
2260 N1=0
2270 GOTO 2010
2280 REM Routine voor het aangeven van
         foutieve invoer
2290 PRINT:PRINT TAB(11);:A$="foutieve i
nvoer!":A=0:GOSUB 2490:GOSUB 2320:RETURN
2300 A$=INKEY$:IF A$="" THEN GOTO 2300 E
LSE RETURN
2310 REM Routine voor het stilzetten
         van de beeldscherm weergave
```

```
2320 PRINT:PRINT TAB(6) "<Druk een toets
 a.u.b.>"
2330 A$=INKEY$:IF A$="" THEN GOTO 2330 E
LSE RETURN
2340 REM Routine algemeen
2350 PRINT "Terug naar het menu d.m.v. <
ENTER>       zonder invoer."
2360 RETURN
2370 REM Het veranderen van >chr$(127)
         -> negatieve karakters
2380 FOR M=2304 TO 3055
2390 A$=BIN$(VPEEK(M))
2400 A$=STRING$(8-LEN(A$),"Ø")+A$
2410 FOR N=1 TO 8
2420 IF MID$(A$,N,1)="1" THEN MID$(A$,N,
1)="Ø" ELSE MID$(A$,N,1)="1"
2430 NEXT N
2440 A$="&b"+A$
2450 VPOKE M+768,VAL(A$)
2460 A$=""
2470 NEXT M
2480 RETURN
2490 REM Veranderen in negatief
2500 REM a=Ø -> alles
2510 REM a=1 -> tot aan de spatie
2520 REM a=2 -> alleen kleine letters
2530 REM a=3 -> alleen hoofdletters
2540 REM a=4 -> alleen cijfers
2550 FOR TQ=1 TO LEN(A$)
2560 IF A=Ø THEN GOTO 261Ø
2570 IF A=1 AND MID$(A$,TQ,1)=" " THEN G
OTO 262Ø
2580 IF A=2 AND (MID$(A$,TQ,1)<CHR$(97)
OR MID$(A$,TQ,1)>CHR$(125)) THEN GOTO 26
2Ø
2590 IF A=3 AND (MID$(A$,TQ,1)<CHR$(65)
OR MID$(A$,TQ,1)>CHR$(93)) THEN GOTO 262
Ø
```

```
2600 IF A=4 AND (MID$(A$,YQ,1)>CHR$(57)
OR MID$(A$,TQ,1)<CHR$(48)) THEN GOTO 262
0
2610 MID$(A$,TQ,1)=CHR$(ASC(MID$(A$,TQ,1
))+96)
2620 NEXT TQ
2630 PRINT A$
2640 RETURN
```

Kalender

Diegenen onder u die reeds gewend zijn om met computers om te gaan, zullen dit programma 'kalender', in deze of in een andere vorm, wel vaker tegenkomen. Zowel afzonderlijk, zoals in dit geval, of als onderdeel van een groter programma, zoals bijvoorbeeld een termijnberekening.

In elk van deze gevallen gaat het om het optellen van jaren, maanden en dagen, waarbij als eindresultaat de gewenste dag van de week berekend wordt.

Het afgebeelde programma bestaat uit drie delen, die als gevolg van de ingevoerde 'INPUT' tot het gewenste jaar, maand en dag zullen leiden.

De opgave van het jaar moet tussen de 1901 en 2001 liggen. Schrikkeljaren worden automatisch geregistreerd. Zie regel 140.

Bij de berekening van het ingevoerde maandgetal wordt de bijbehorende naam toegevoegd en het maximaal aantal dagen van die betreffende maand vastgesteld. Wordt een zgn. 'flag', ofwel markering, in de jaaropgave in de vorm van Z=1 = schrikkeljaar en M=2 = februari geconstateerd, dan wordt ook de 29ste februari in verband met het schrikkeljaar meegeteld. Zie regel 590.

Hierna volgt de verwijzing naar de afzonderlijke weekdagen. Zie regel 750. En ten slotte wordt de uitkomst afgebeeld. ('De . . . van het jaar . . . is een . . .'). Indien gewenst kan men weer opnieuw beginnen en een andere dag laten uitrekenen.

```
10 REM Kalender
20 REM MSX Verzamelde programma's
30 REM Copyright 1984 DATA BECKER &
                      Rainer Lueers
40 SCREEN 0:COLOR 15,1,15:KEY OFF
50 WIDTH 40
60 PRINT
70 PRINT
80 PRINT
90 PRINT "        Gewenste dag v.d. week
                  van 1901 t/m 2000"
100 PRINT:PRINT
110 REM Invoer van het jaartal en de
        berekening (ook schrikeljaar)
120 INPUT "Jaar  ";J
130 IF J<1901 OR J>2000 THEN GOSUB 990:G
OTO 50
140 IF INT(J/4)=J/4 THEN Z=1 ELSE Z=0
150 J1=2
160 IF J=1901 THEN GOTO 230
170 FOR N=1902 TO J
180 J1=J1+1
190 IF INT((N-1)/4)=(N-1)/4 THEN J1=J1+1
200 IF J>6 THEN J1=J1-7
210 NEXT N
220 REM Invoer van de maand en berekenen
        van (de naam en het aantal dagen
230 INPUT "Maand ";M
240 IF M<1 OR M>12 THEN GOSUB 990:GOTO 5
0
250 ON M GOSUB 270,290,310,330,350,370,3
90,410,430,450,470,490
260 GOTO 510
270 M$="januari"
280 RETURN
290 M$="februari"
300 RETURN
310 M$="maart"
320 RETURN
```

```
330 M$="april"
340 RETURN
350 M$="mei"
360 RETURN
370 M$="juni"
380 RETURN
390 M$="juli"
400 RETURN
410 M$="augustus"
420 RETURN
430 M$="september"
440 RETURN
450 M$="oktober"
460 RETURN
470 M$="november"
480 RETURN
490 M$="december"
500 RETURN
510 RESTORE:FOR N=1 TO M
520 READ M1
530 NEXT N
540 DATA 31,28,31,30,31,30
550 DATA 31,31,30,31,30,31
560 IF Z=1 AND M=2 THEN M1=29
570 REM Invoer van de dag (getal) en
        berekening v.d. dag v.d. week
580 INPUT "Dag    ";T
590 IF T=29 AND M=2 AND Z<>1 THEN GOSUB
990:GOTO 50
600 IF T<1 OR T>M1 THEN GOSUB 990:GOTO 5
0
610 T1=0
620 RESTORE
630 IF M=1 THEN GOTO 690
640 FOR N=2 TO M
650 READ M1
660 T1=T1+M1
670 NEXT N
```

```
68Ø IF Z=1 AND M>2 THEN T1=T1+1
69Ø T1=T1+T+J1
7ØØ T1=T1-INT(T1/7)*7
71Ø IF T1=Ø THEN T1=7
72Ø PRINT
73Ø PRINT "          De";T;"e ";M$;" in het
 jaar";J
74Ø PRINT "          is/was een ";
75Ø ON T1 GOSUB 84Ø,86Ø,88Ø,9ØØ,92Ø,94Ø,
96Ø
76Ø PRINT
77Ø PRINT
78Ø INPUT"       Wilt u nog een dag
                 laten uitrekenen ( /n) ";
F$
79Ø IF LEFT$(F$,1)="N" OR LEFT$(F$,1)="n
" THEN F$="N"
8ØØ IF F$="N" THEN END
81Ø RESTORE
82Ø CLS
83Ø GOTO 6Ø
84Ø PRINT "zondag"
85Ø RETURN
86Ø PRINT "maandag"
87Ø RETURN
88Ø PRINT "dinsdag"
89Ø RETURN
9ØØ PRINT "woensdag"
91Ø RETURN
92Ø PRINT "donderdag"
93Ø RETURN
94Ø PRINT "vrijdag"
95Ø RETURN
96Ø PRINT "zaterdag"
97Ø RETURN
98Ø END
```

```
              Gewenste dag v.d. week
              van 1901 t/m 2000

Jaar  ? 1984
Maand ? 12
Dag   ? 26

       De 26e december in het jaar 1984
             is/was een woensdag

       Wilt u nog een dag
       laten uitrekenen ( /n) ?
```

Balkendiagram

Arne kreeg onlangs een speciale opgave op school, die zowel een uitdaging als een stimulans zou blijken te zijn voor zijn programmeerwoede en zijn enthousiasme voor grafische afbeeldingen. Als resultaat van een sociologische studie waren de nodige gegevens verkregen en deze moesten nu – als het even kon – als balkendiagram in beeld gebracht worden, liefst net zo als bij de televisie-uitzending van Brandpunt in de markt.

Voorwaar een opgave die, alvorens ze gerealiseerd kon worden, voor Arne al gauw de nodige zweetdruppeltjes betekende. Maar, zoals u ziet, het is ten slotte toch goed gelukt!

Voor de vele programmeerfreaks in den lande lijkt het misschien op het eerste gezicht helemaal niet zo bijzonder dat een scholier ook eens een dergelijke opgave moet oplossen. Maar als men weet dat Arne een (pas) 12-jarige gymnasiast uit Munster is, dan moet men hem niet de oplossing van dit probleem toch wel bijzonder hartelijk gelukwensen.

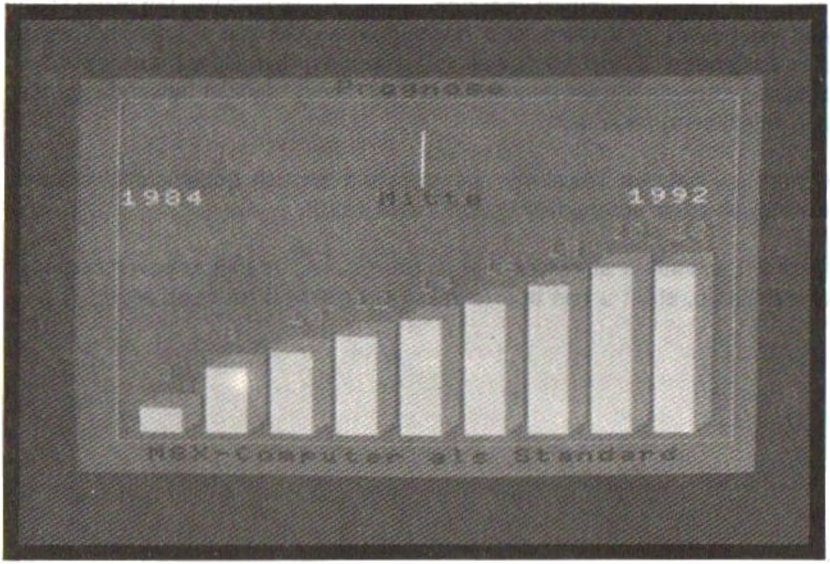
Omdat u, beste lezer, weinig aan de sociologische gegevens van Arne heeft, heb ik het programma een klein beetje omgewerkt en het op een algemeen toepasbaar niveau gebracht. Gebleven zijn in ieder geval de beperkingen die Arne, met zijn schoolproject voor ogen, in dit programma van het begin af aan had ingebouwd. Er kunnen maar negen balken getekend worden, die slechts met de getallen '0' t/m '20' benoemd kunnen worden. Daar staat tegenover, zoals de foto van het beeldscherm laat zien, een glashelder balkendiagram en nog driedimensionaal ook.

U kunt uw getallen zowel van een opschrift als een onderschrift voorzien, waarbij de tekst automatisch gecentreerd wordt.

Voor het schriftelijk afbeelden van de links en rechts aan te geven begin- en eindwaarden, staat u een aparte invoermogelijkheid ter beschikking.

Een variatie op het aantal af te beelden waarden is niet eenvoudig aan te brengen. Vandaar de nu volgende tip voordat u begint het programma aan uw eigen wensen aan te passen. De afbeeldingsnormen (voor het balkendiagram) lopen in het programa via 'FOR . . . NEXT'-lussen. Hierbij wordt iedere waarde eenmaal doorlopen. ('STEP 1'). Verandert u dit echter in 'STEP 2' dan wordt telkens slechts elke tweede ingevoerde waarde voor afbeelding op het scherm gebruikt. Dus de waarden 1,3,5,7 en 9. Hierdoor worden in dat geval slechts vijf balken afgebeeld.

Wij wensen u veel plezier toe bij het omzetten van uw getallen in een driedimensionaal balkendiagram. Arne en de auteur Luers.



```
10 REM Balkendiagram
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
    Rainer Lueers & Arne Stoffregen
40 SCREEN 0:COLOR15,1,1:WIDTH 40
50 REM Invoer van de nieuwe waarden
        het opschrift en de
        klassificaties
60 INPUT "Opschrift (max. 30 karakters)
         ";C$:IF LEN(C$)>30 THEN GOTO 60
70 INPUT "Ondertitel (max. 30 karakters)
         ";D$:IF LEN(D$)>30 THEN GOTO 70
80 INPUT "Begin-waarde (b.v. pos.) ";A$:
IF LEN(A$)>6 THEN A$=LEFT$(A$,6)
90 INPUT "Eind-waarde (b.v. neg.) ";B$:I
F LEN(B$)>6 THEN B$=LEFT$(B$,6)
100 PRINT:PRINT "Invoer van 9 waarden:":
PRINT
110 FOR N=1 TO 9:PRINT "waarde";N;:INPUT
 " ( 0 tot 20) ";WT(N)
120 IF WT(N)<0 OR WT(N)>20 THEN GOTO 110
 ELSE NEXT N
130 COLOR 1,13:OPEN "grp:"FOR OUTPUT AS
#1
140 ON STOP GOSUB 570:STOP ON
150 FOR T=1 TO 9 STEP 1
160 C=C+WT(T)
170 IF WT(T)=0 THEN H(T)=.8*4:GOTO 190
180 H(T)=WT(T)*4
190 NEXT T
200 FH=170
210 REM Grafische weergave v.d. tekst
        en het beeldmasker
220 SCREEN 2
230 LINE(15,10)-(246,175),14,B
240 LINE(127,25)-(127,50),15
250 DRAW "bm 112,53"
260 PRINT #1,"Centr."
270 DRAW "bm 42,1"
280 COLOR 1
```

```
290 A=(255-(LEN(C$)*8))/2:DRAW "bm =a;,1
"
300 REM Weergave van het opschrift en
        onderschrift - in het midden
        gecentreerd
310 PRINT #1,C$
320 A=(255-(LEN(D$)*8))/2:DRAW "bm =a;,1
78":PRINT #1,D$:COLOR 15
330 DRAW "bm 17,53"
340 PRINT #1,A$
350 A=230-LEN(B$)*6:DRAW "bm=a;,53;"
360 PRINT #1,B$
370 REM Grafisch weergeven van de
        ingevoerde waardes in de
        vorm van een balkendiagram
380 FOR I=1 TO 9 STEP 1
390 LINE (I*24,FH-H(I))-(I*24+15,FH),15,
BF
400 LINE -(I*24+21,FH-6),14
410 LINE -(I*24+21,FH-6-H(I)),14
420 LINE -(I*24+15,FH-H(I)),14
430 LINE -(I*24+15,FH),14
440 PAINT (I*24+17,FH-H(I)+2),14
450 LINE (I*24+21,FH-6-H(I))-(I*24+9,FH-
6-H(I)),14
460 LINE -(I*24,FH-H(I)),14
470 LINE (I*24,FH-H(I))-(I*24+15,FH-H(I)
),14
480 PAINT (I*24+8,FH-H(I)-4),14
490 NEXT I
500 FOR T=1 TO 9 STEP 1
510 A=T*24-2:B=FH-H(T)-20:DRAW "bm =a;,=
b;"
520 COLOR 3
530 PRINT #1,WT(T)
540 NEXT T
550 FOR T=1 TO 5000:NEXT T
560 A$=INKEY$:IF A$="" THEN GOTO 560
570 COLOR 15,1
```

Sporttabel

Aan dit programma kunt u in de toekomst heel wat plezier beleven. Bovendien kan het u de nodige tijd besparen.

Hierbij wordt door de computer precies dat berekend, wat bij sportverenigingen per jaar zo vaak en regelmatig moet gebeuren. Omdat voetbal als karakterestiek voorbeeld kan worden gezien, werd dit dan ook als voorbeeld in ons programma gebruikt.

Als eerste invoer kunt u het aantal verenigingen bepalen. Minimaal vier maximaal twintig. Aansluitend wordt u de vraag gesteld of u de namen van de clubs die reeds in de DATA-regels werden vastgelegd, voor het te berekenen seizoen wilt gebruiken (de eerste en eredivisie), of zelf maximaal twintig clubs wilt invoeren.

Ook de wedstrijden van de diverse clubs in de eerste en tweede helft van het seizoen zijn reeds in alle mogelijke combinaties voor uit en thuis vastgelegd. Wilt u deze gegevens vanuit het computergeheugen overnemen, dan beantwoordt u ook deze vraag met 'j'(a).

Hierbij een tip. De in de DATA-regels vastgelegde wedstrijden resulteren in een willekeurige samenstelling, waarbij ten slotte iedere club tegen alle clubs gespeeld heeft. Eenmaal thuis en eenmaal uit. Als u de gegevens uit de DATA-regels niet wilt overnemen, dan dient u, overeenkomstig het afgebeelde menu, achtereenvolgens alle speeldagen in het geheugen in te voeren. U hoeft in dit geval maar de helft van het seizoen in te voeren. De andere helft berekent uw MSX-computer voor u.

Aansluitend wordt u een menu getoond dat u de keuze uit drie mogelijkheden biedt: ' 1. speeldag invoeren 2. huidige tabel tonen 3. speeldag tonen'.

Laten we eerst keuzemogelijkheid 1 maar eens kiezen. Omdat het gebeuren kan dat de een of andere wedstrijd uitvalt, is ook hierin voorzien. Het invoeren van '99,99' betekent spel uitgevallen.

Als de tweede speeldag nog niet heeft plaatsgevonden en de reeds ingevoerde resultaten onze voorkeur hebben, kunnen we nu onze gegevens in een overzichtelijke vorm op de monitor laten afbeelden. Hoe dat werkt? '3. speeldag tonen'. Als we voor deze keuzemogelijkheid kiezen, wordt ons de mogelijkheid geboden het hele seizoen door een druk op een toets voor onze ogen te laten weergeven. We kunnen echter ook het resultaat per dag laten afbeelden.

Kijken we naar de tweede helft van het seizoen, bij twintig clubs vanaf de twintigste wedstrijd, dan wordt ons, indien aanwezig, ook het spel van de eerste helft op het beeldscherm getoond.

Zijn er bijvoorbeeld door het slechte weer wedstrijden uitgevallen (invoer '99,99'), dan kunnen we deze alsnog invoeren. Hiervoor dient u weer '1. speeldag invoeren' te kiezen en de overeenkomstige speeldag op te vragen, waarvan de een of andere wedstrijd nog openstaat. Uw MSX-computer zal met een kort piepje de reeds benoemde wedstrijden passeren en precies daar stoppen waar nog gegevens ingevoerd moeten worden. Zijn er meerdere wedstrijden op een dag uitgevallen, dan moeten de nog niet ingehaalde wedstrijden opnieuw met '99,99' benoemd worden.

Thans uw aandacht voor de essentie van dit programma, namelijk het berekenen van de tabellen. Het duurt even tot de tabel verschijnt. Maar ze is dan ook op rangorde en bovendien in keurige kolommen samengesteld. Achter het bereikte plaatsingsnummer en de afgekorte naam van de club, wordt het aantal van de reeds gespeelde wedstrijden van elke club afgebeeld. Ten slotte wordt ook nog het huidige doelgemiddelde en het totaal aantal punten in beeld gebracht. Het laatstgenoemde aangevuld met het positieve of negatieve verschil in plus- of mintekens.

Zo blijft het overzicht altijd gewaarborgd. Hoe en met welk resultaat wanneer gespeeld werd en wat de plaats van welke club is op dat moment.

Een goed bruikbaar programma, dat maar één ding van u eist. Zet uw computer niet uit! Want het saven van de gegevens is nog niet ingebouwd. Dit omdat het met gebruik van een cassette eenvoudig te lang zou duren om zoveel gegevens te verwerken. Laten we dus maar geduldig wachten op een diskdrive! Het aanpassen zal u niet te veel moeilijkheden opleveren, omdat we van het begin af aan de gedimensioneerde variabelen met een REM-regel verklaard hebben. Daarentegen kunt u zonder meer een lijst laten uitprinten van de tabellen, als u de aanwezige PRINT-commando's (regels 900 tot 960) in een LPRINT-commando verandert.

Speeldag 20

1	Duisburg	Hannover	1	:	1	7	:	1
2	Utrecht	MVV	2	:	2	3	:	3
3	Roda	Haarlem	3	:	3	4	:	5
4	Volendam	Sittard	4	:	4	-	:	-
5	Den Bosch	Volendam	3	:	3	-	:	-
6	Ulm	Aachen	6	:	2	3	:	4
7	Freiburg	Kassel	3	:	5	5	:	4
8	Hertha	Solingen	4	:	6	6	:	3

Druk op een toets s.v.p.

```
10 REM Sporttabel
20 REM MSX Verzamelde programma's
30 REM Copyright 1984 DATA BECKER &
                          Rainer Lueers
40 SCREEN 0:KEY OFF:COLOR 15,1,1:PRINT "
Een ogenblikje a.u.b.":GOSUB 61000:CLS
50 WIDTH 40:DEFINT A-Z
60 REM Keuze tussen 4 t/m 20 clubs
70 INPUT "Aantal clubs (4/6/8/10/12/14/1
6/18/20) ";F$
80 REM Lijst v.d. variabelen die
       gedimensioneerd worden met
       hun betekenis
90 REM a$( )   = namen clubs
       b( , ) = wedstrijd gespeeld
       c( , ) = resultaat
       e( , ) = totaal score
       f( , ) = totaal tegendoelpunten
100 REM g( , ) = totaal gewonnen
        h( , ) = totaal gelijk
        i( , ) = totaal verloren
        k( , , ) = competitie combin.
        z1 tot z5= voor sorteerroutine
110 F=VAL(F$):IF F/2<>INT(F/2) THEN PRIN
T:GOSUB 1300:GOTO 50
120 IF F<4 OR F>20 THEN GOSUB 1300:GOTO
50
130 DIM A$(F),B(F*2,F/2),C(F,F*2-2,2),K(
F-1,F/2,2),E(F),F(F),G(F),H(F),I(F),Z1(F
+1),Z2(F+1),Z3(F+1),Z4(F+1),Z5(F+1)
140 REM Invoer van namen van clubs
        of inlezen van clubs vanuit
        DATA-regels
150 INPUT "Namen vanuit programma inleze
n (J/ )";F$
160 IF LEFT$(F$,1)="j" OR LEFT$(F$,1)="J
" THEN F$="J"
170 IF F$="J" THEN GOTO 240
180 CLS
190 FOR N=1 TO F
```

```
200 PRINT "Club";N;:INPUT ": ";F$
210 IF LEN(F$)>10 THEN A$(N)=LEFT$(F$,10
) ELSE A$(N)=F$
220 NEXT N
230 GOTO 360
240 ON F GOSUB 260,260,260,260,260,270,2
60,280,260,290,260,300,260,310,260,320,2
60,330,260,340
250 GOTO 360
260 RESTORE 6350:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
270 RESTORE 6320:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
280 RESTORE 6290:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
290 RESTORE 6260:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
300 RESTORE 6230:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
310 RESTORE 6200:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
320 RESTORE 6170:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
330 RESTORE 6130:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
340 RESTORE 6080:FOR N=1 TO F:READ A$(N)
:NEXT N:RETURN
350 REM Competitievolgorde in DATA-
        regels (demo) kan gebruikt
        worden. Het is ook mogelijk
        alle wedstrijden zelf in te
        voeren
360 INPUT "Competitievolgorde uit prg (J
/ ) ";F$:IF LEFT$(F$,1)="j" OR LEFT$(F$,
1)="J" THEN F$="J"
370 IF F$="J" THEN GOTO 440
380 PRINT:PRINT:PRINT "Wedstrijdvolgorde
 met getallen invoeren          (b.v. 1 <c
omma> 2 <ENTER>          Vorige invoer h
erhalen -> 0 invoeren"
```

```
390 FOR N=1 TO F-1:PRINT:PRINT:PRINT "We
dstrijddag";N:PRINT
400 FOR M=1 TO F/2
410 PRINT:PRINT "Wedstrijd";M:INPUT A,B:
IF A>F OR B>F OR A=B THEN GOTO 410 ELSE
IF A=0 OR B=0 THEN M=M-1:GOTO 410 ELSE K
(N,M,1)=A:K(N,M,2)=B:A=0:B=0:PRINT A$(K(
N,M,1));" - ";A$(K(N,M,2))
420 NEXT M,N
430 GOTO 550
440 ON F GOSUB 460,460,460,460,460,470,4
60,480,460,490,460,500,460,510,460,520,4
60,530,460,540
450 GOTO 550
460 RESTORE 5000:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
470 RESTORE 5040:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
480 RESTORE 5100:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
490 RESTORE 5180:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
500 RESTORE 5280:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
510 RESTORE 5400:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
520 RESTORE 5540:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
530 RESTORE 5700:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
```

```
540 RESTORE 5880:FOR N=1 TO F-1:FOR M=1
TO F/2:READ K(N,M,1),K(N,M,2):NEXT M,N:R
ETURN
550 CLS:A$="Sportclub-tabel":A=0:LOCATE
6,6:GOSUB 61120
560 LOCATE 6,9
570 PRINT "1. Wedstrijddag invoeren"
580 LOCATE 6,10
590 PRINT "2. Huidige tabel laten zien"
600 LOCATE 6,11
610 PRINT "3. Wedstrijddag bekijken
620 LOCATE 6,13
630 INPUT " Uw keuze (1 t/m 3) ";F$
640 IF VAL(F$)<1 OR VAL(F$)>4 THEN GOSUB
 1280:GOSUB 1300:GOTO 440
650 ON VAL(F$) GOSUB 680,920,1010
660 GOTO 550
670 REM Wedstrijddag invoeren
680 CLS
690 INPUT "Welke wedstrijddag ";F$:IF VA
L(F$)<1 OR VAL(F$)>F*2-2 THEN GOSUB 1300
:GOTO 690
700 IF VAL(F$)>F-1 THEN GOTO 760
710 FOR N=1 TO F/2:PRINT"wedstrijd";N
720 IF B(VAL(F$),N)<>0 THEN PRINT CHR$(7
);"Fout! Spelresultaat bekent:":PRINT A$
(K(VAL(F$),N,1));"-";A$(K(VAL(F$),N,2));
":";C(K(VAL(F$),N,1),VAL(F$),1);"-";C(K(
VAL(F$),N,1),VAL(F$),2):GOTO 750
730 PRINT A$(K(VAL(F$),N,1));" - ";A$(K(
VAL(F$),N,2));:INPUT A,B:IF A=99 THEN PR
INT "uitgevallen":B(VAL(F$),N)=0:GOTO 75
0 ELSE GOSUB 810
740 B(VAL(F$),N)=1:C(K(VAL(F$),N,1),VAL(
F$),1)=A:C(K(VAL(F$),N,1),VAL(F$),2)=B:C
(K(VAL(F$),N,2),VAL(F$),1)=B:C(K(VAL(F$)
,N,2),VAL(F$),2)=A
750 NEXT N:RETURN
760 FOR N=1 TO F/2:PRINT "Wedstrijd";N
```

```
770 IF B(VAL(F$),N)<>Ø THEN PRINT CHR$(7
);"FOUT! Spelresultaat bekent:":PRINT A$
(K(VAL(F$)-F+1,N,2));"-";A$(K(VAL(F$)-F+
1,N,1));":";C(K(VAL(F$)-F+1,N,1),VAL(F$)
,1);"-";C(K(VAL(F$)-F+1,N,1),VAL(F$),2):
GOTO 8ØØ
78Ø PRINT A$(K(VAL(F$)-F+1,N,2));" - ";A
$(K(VAL(F$)-F+1,N,1));:INPUT A,B:IF A=99
 THEN PRINT "uitgevallen":B(VAL(F$),N)=Ø
:GOTO 8ØØ ELSE GOSUB 86Ø
79Ø B(VAL(F$),N)=1:C(K(VAL(F$)-F+1,N,1),
VAL(F$),1)=A:C(K(VAL(F$)-F+1,N,1),VAL(F$
),2)=B:C(K(VAL(F$)-F+1,N,2),VAL(F$),1)=B
:C(K(VAL(F$)-F+1,N,2),VAL(F$),2)=A
8ØØ NEXT N: RETURN
81Ø E(K(VAL(F$),N,1))=E(K(VAL(F$),N,1))+
A:F(K(VAL(F$),N,1))=F(K(VAL(F$),N,1))+B:
E(K(VAL(F$),N,2))=E(K(VAL(F$),N,2))+B:F(
K(VAL(F$),N,2))=F(K(VAL(F$),N,2))+A
82Ø IF A=B THEN H(K(VAL(F$),N,1))=H(K(VA
L(F$),N,1))+1:H(K(VAL(F$),N,2))=H(K(VAL(
F$),N,2))+1
83Ø IF A>B THEN G(K(VAL(F$),N,1))=G(K(VA
L(F$),N,1))+2:I(K(VAL(F$),N,2))=I(K(VAL(
F$),N,2))+2
84Ø IF A<B THEN I(K(VAL(F$),N,1))=I(K(VA
L(F$),N,1))+2:G(K(VAL(F$),N,2))=G(K(VAL(
F$),N,2))+2
85Ø RETURN
86Ø E(K(VAL(F$)-F+1,N,1))=E(K(VAL(F$)-F+
1,N,1))+A:F(K(VAL(F$)-F+1,N,1))=F(K(VAL(
F$)-F+1,N,1))+B:E(K(VAL(F$)-F+1,N,2))=E(
K(VAL(F$)-F+1,N,2))+B:F(K(VAL(F$)-F+1,N,
2))=F(K(VAL(F$)-F+1,N,2))+A
87Ø IF A=B THEN H(K(VAL(F$)-F+1,N,1))=H(
K(VAL(F$)-F+1,N,1))+1:H(K(VAL(F$)-F+1,N,
2))=H(K(VAL(F$)-F+1,N,2))+1
88Ø IF A>B THEN G(K(VAL(F$)-F+1,N,1))=G(
K(VAL(F$)-F+1,N,1))+2:I(K(VAL(F$)-F+1,N,
2))=I(K(VAL(F$)-F+1,N,2))+2
```

```
890 IF A<B THEN I(K(VAL(F$)-F+1,N,1))=I(
K(VAL(F$)-F+1,N,1))+2:G(K(VAL(F$)-F+1,N,
2))=G(K(VAL(F$)-F+1,N,2))+2
900 RETURN
910 REM Huidige tabel tonen
920 CLS:GOSUB 1190
930 A$="Pl. Club   Sp. Doelg.    punten
 <>   ":A=0:GOSUB 61120
940 FOR N=1 TO F
950 PRINT N;TAB(4);LEFT$(A$(Z5(N)),5);TA
B(10);(G(Z5(N))+(H(Z5(N))*2)+I(Z5(N)))/2
;TAB(14);E(Z5(N));TAB(19);":";F(Z5(N));T
AB(25);G(Z5(N))+H(Z5(N));TAB(29);":"H(Z5
(N))+I(Z5(N));TAB(34);G(Z5(N))-I(Z5(N))
960 NEXT N
970 A$="       <Druk een toets a.u.b.>
      ":A=0:GOSUB 61120
980 A$=INKEY$:IF A$="" THEN GOTO 980
990 RETURN
1000 REM Wedstrijddag tonen
1010 CLS:PRINT:INPUT "Het hele seizoen (
J/ ) ";F$:IF LEFT$(F$,1)="j" OR LEFT$(F$
,1)="J" THEN F$="J"
1020 IF F$="J" THEN GOTO 1110
1030 PRINT "Wedstrijddag ( 1 tot";F*2-2;
:INPUT ") ";F$:IF VAL(F$)<1 OR VAL(F$)>F
*2-2 THEN GOSUB 1300 GOTO 1030 ELSE IF V
AL(F$)>F-1 THEN GOTO 1070
1040 N=VAL(F$):CLS:PRINT "Wedstrijddag";
N:PRINT:PRINT:PRINT:FOR M=1 TO F/2:PRINT
 M;TAB(4);A$(K(N,M,1));TAB(14);A$(K(N,M,
2));
1050 IF B(N,M)=0 THEN PRINT TAB(32);"-
: -" ELSE PRINT TAB(31);C(K(N,M,1),N,1);
TAB(35);":";TAB(36);C(K(N,M,1),N,2);
1060 NEXT M:PRINT:PRINT:GOSUB 1280:GOTO
1180
1070 N=VAL(F$):CLS:PRINT "wedstrijddag";
N:N=N-(F-1):PRINT:PRINT:PRINT:FOR M=1 TO
 F/2:PRINT M;TAB(4);A$(K(N,M,2));TAB(13)
;A$(K(N,M,1));TAB(22);
```

```
1080 IF B(N,M)=1 THEN PRINT C(K(N,M,1),N
,2);TAB(26);":";TAB(27);C(K(N,M,1),N,1);
 ELSE PRINT " -  : -";
1090 IF B(N+(F-1),M)=0 THEN PRINT TAB(32
);"-  : -" ELSE PRINT TAB(31);C(K(N,M,1)
,N+F-1,1);TAB(35);":";TAB(36);C(K(N,M,1)
,N+F-1,2);
1100 NEXT M:PRINT:PRINT:GOSUB 1280:GOTO
1180
1110 FOR N=1 TO F-1:CLS:PRINT "Wedstrijd
dag" ;N:PRINT:PRINT:PRINT:FOR M=1 TO F/2
:PRINT;M;TAB(4);A$(K(N,M,1));TAB(14);A$(
K(N,M,2));
1120 IF B(N,M)=0 THEN PRINT TAB(32);" -
: -"; ELSE PRINT TAB(32);C(K(N,M,1),N,1)
;TAB(35);":";TAB(36);C(K(N,M,1),N,2);
1130 NEXT M:PRINT:PRINT:GOSUB 1280 NEXT
N
1140 FOR N=1 TO F-1:CLS:PRINT "Wedstrijd
dag";N+F-1:PRINT:PRINT:PRINT:FOR M=1 TO
F/2:PRINT M;TAB(4);A$(K(N,M,2));TAB(13);
A$(K(N,M,1));TAB(22);
1150 IF B(N,M)=0 THEN PRINT " -  : -"; E
LSE PRINT C(K(N,M,1),N,2);TAB(26);":";TA
B(28);C(K(N,M,1),N,1);
1160 IF B(N+F-1,M)=0 THEN PRINT TAB(32);
"-  : -" ELSE PRINT TAB(31);C(K(N,M,1),N
+F-1,1);TAB(35);":";TAB(36);C(K(N,M,1),N
+F-1,2);
1170 NEXT M:PRINT:PRINT:GOSUB 1280:NEXT
1180 RETURN
1190 REM Sorteer-subroutine
1200 FOR N=1 TO F:Z1(N)=G(N)+H(N):Z2(N)=
E(N)-F(N):Z3(N)=E(N):Z4(N)=-1:NEXT N
1210 FOR N=1 TO F
1220 FOR M=1 TO F
1230 IF Z1(M)>Z4(N) THEN Z4(N)=Z1(M):A=M
:GOTO 1260
1240 IF Z1(M)=Z4(N) AND Z2(M)>Z2(A) THEN
 Z4(N)=Z1(M):A=M:GOTO 1260
```

```
1250 IF Z1(M)=Z4(N) AND Z2(M)=Z2(A) AND
Z3(M)>Z3(A) THEN Z4(N)=Z1(M):A=M
1260 NEXT M:Z1(A)=-1:Z5(N)=A:A=F+1:NEXT
N
1270 RETURN
1280 PRINT:PRINT TAB(6) "<Druk een toets
 a.u.b.>"
1290 A$=INKEY$:IF A$="" THEN GOTO 1290 E
LSE RETURN
1300 PRINT TAB(10) "Verkeerde invoer!":G
OSUB 1280:RETURN
5000 REM Data voor vier clubs
5010 DATA 1,4,2,3
5020 DATA 4,3,1,2
5030 DATA 2,4,3,1
5040 REM Data voor zes clubs
5050 DATA 1,6,2,5,3,4
5060 DATA 6,4,5,3,1,2
5070 DATA 2,6,3,1,4,5
5080 DATA 6,5,1,4,2,3
5090 DATA 3,6,4,2,5,1
5100 REM Data voor acht clubs
5110 DATA 1,8,2,7,3,6,4,5
5120 DATA 8,5,6,4,7,3,1,2
5130 DATA 2,8,3,1,4,7,5,6
5140 DATA 8,6,7,5,1,4,2,3
5150 DATA 3,8,4,2,5,1,6,7
5160 DATA 8,7,1,6,2,5,3,4
5170 DATA 4,8,5,3,6,2,7,1
5180 REM data voor tien clubs
5190 DATA 1,10,2,9,3,8,4,7,5,6
5200 DATA 10,6,7,5,8,4,9,3,1,2
5210 DATA 2,10,3,1,4,9,5,8,6,7
5220 DATA 10,7,8,6,9,5,1,4,2,3
5230 DATA 3,10,4,2,5,1,6,9,7,8
5240 DATA 10,8,9,7,1,6,2,5,3,4
5250 DATA 4,10,5,3,6,2,7,1,8,9
5260 DATA 10,9,1,8,2,7,3,6,4,5
5270 DATA 5,10,6,4,7,3,8,2,9,1
5280 REM Data voor twaalf clubs
```

```
5290 DATA 1,12,2,11,3,10,4,9,5,8,6,7
5300 DATA 12,7,8,6,9,5,10,4,11,3,1,2
5310 DATA 2,12,3,1,4,11,5,10,6,9,7,8
5320 DATA 12,8,9,7,10,6,11,5,1,4,2,3
5330 DATA 3,12,4,2,5,1,6,11,7,10,8,9
5340 DATA 12,9,10,8,11,7,1,6,2,5,3,4
5350 DATA 4,12,5,3,6,2,7,1,8,11,9,10
5360 DATA 12,10,11,9,1,8,2,7,3,6,4,5
5370 DATA 5,12,6,4,7,3,8,2,9,1,10,11
5380 DATA 12,11,1,10,2,9,3,8,4,7,5,6
5390 DATA 6,12,7,5,8,4,9,3,10,2,11,1
5400 REM Data voor veertien clubs
5410 DATA 1,14,2,13,3,12,4,11,5,10,6,9,7
,8
5420 DATA 14,8,9,7,10,6,11,5,12,4,13,3,1
,2
5430 DATA 2,14,3,1,4,13,5,12,6,11,7,10,8
,9
5440 DATA 14,9,10,8,11,7,12,6,13,5,1,4,2
,3
5450 DATA 3,14,4,2,5,1,6,13,7,12,8,11,9,
10
5460 DATA 14,10,11,9,12,8,13,7,1,6,2,5,3
,4
5470 DATA 4,14,5,3,6,2,7,1,8,13,9,12,10,
11
5480 DATA 14,11,12,10,13,9,1,8,2,7,3,6,4
,5
5490 DATA 5,14,6,4,7,3,8,2,9,1,10,13,11,
12
5500 DATA 14,12,13,11,1,10,2,9,3,8,4,7,5
,6
5510 DATA 6,14,7,5,8,4,9,3,10,2,11,1,12,
13
5520 DATA 14,13,1,12,2,11,3,10,4,9,5,8,6
,7
5530 DATA 7,14,8,6,9,5,10,4,11,3,12,2,13
,1
5540 REM Data voor zestien clubs
```

```
5550 DATA 1,16,2,15,3,14,4,13,5,12,6,11,
7,10,8,9
5560 DATA 16,9,10,8,11,7,12,6,13,5,14,4,
15,3,1,2
5570 DATA 2,16,3,1,4,15,5,14,6,13,7,12,8
,11,9,10
5580 DATA 16,10,11,9,12,8,13,7,14,6,15,5
,1,4,2,5
5590 DATA 3,16,4,2,5,1,6,15,7,14,8,13,9,
12,10,11
5600 DATA 16,11,12,10,13,9,14,8,15,7,1,6
,2,5,3,4
5610 DATA 4,16,5,3,6,2,7,1,8,15,9,14,10,
13,11,12
5620 DATA 16,12,13,11,14,10,15,9,1,8,2,7
,3,6,4,5
5630 DATA 5,16,6,4,7,3,8,2,9,1,10,15,11,
14,12,13
5640 DATA 16,13,14,12,15,11,1,10,2,9,3,8
,4,7,5,6
5650 DATA 6,16,7,5,8,4,9,3,10,2,11,1,12,
15,13,14
5660 DATA 16,14,15,13,1,12,2,11,3,10,4,9
,5,8,6,7
5670 DATA 7,16,8,6,9,5,10,4,11,3,12,2,13
,1,14,15
5680 DATA 16,15,1,14,2,13,3,12,4,11,5,10
,6,9,7,8
5690 DATA 8,16,9,7,10,6,11,5,12,4,13,3,1
,4,2,15,1
5700 REM Data voor achttien clubs
5710 DATA 1,18,2,17,3,16,4,15,5,14,6,13,
7,12,8,11,9,10
5720 DATA 18,10,11,9,12,8,13,7,14,6,15,5
,16,4,17,3,1,2
5730 DATA 2,18,3,1,4,17,5,16,6,15,7,14,8
,13,9,12,10,11
5740 DATA 18,11,12,10,13,9,14,8,15,7,16,
6,17,5,1,4,2,3
```

```
5750 DATA 3,18,4,2,5,1,6,17,7,16,8,15,9,
14,10,13,11,12
5760 DATA 18,12,13,11,14,10,15,9,16,8,17
,7,1,6,2,5,3,4
5770 DATA 4,18,5,3,6,2,7,1,8,17,9,16,10,
15,11,14,12,13
5780 DATA 18,13,14,12,15,11,16,10,17,9,1
,8,2,7,3,6,4,5
5790 DATA 5,18,6,4,7,3,8,2,9,1,10,17,11,
16,12,15,13,14
5800 DATA 18,14,15,13,16,12,17,11,1,10,2
,9,3,8,4,7,5,6
5810 DATA 6,18,7,5,8,4,9,3,10,2,11,1,12,
17,13,16,14,15
5820 DATA 18,15,16,14,17,13,1,12,2,11,3,
10,4,9,5,8,6,7
5830 DATA 7,18,8,6,9,5,10,4,11,3,12,2,13
,1,14,17,15,16
5840 DATA 18,16,17,15,1,14,2,13,3,12,4,1
1,5,10,6,9,7,8
5850 DATA 8,18,9,7,10,6,11,5,12,4,13,3,1
4,2,15,1,16,17
5860 DATA 18,17,1,16,2,15,3,14,4,13,5,12
,6,11,7,10,8,9
5870 DATA 9,18,10,8,11,7,12,6,13,5,14,4,
15,3,16,2,17,1
5880 REM Data voor twintig clubs
5890 DATA 1,20,2,19,3,18,4,17,5,16,6,15,
7,14,8,13,9,12,10,11
5900 DATA 20,11,12,10,13,9,14,8,15,7,16,
6,17,5,18,4,19,3,1,2
5910 DATA 2,20,3,1,4,19,5,18,6,17,7,16,8
,15,9,14,10,13,11,12
5920 DATA 20,12,13,11,14,10,15,9,16,8,17
,7,18,6,19,5,1,4,2,3
5930 DATA 3,20,4,2,5,1,6,19,7,18,8,17,9,
16,10,15,11,14,12,13
5940 DATA 20,13,14,12,15,11,16,10,17,9,1
8,8,19,7,1,6,2,5,3,4
5950 DATA 4,20,5,3,6,2,7,1,8,19,9,18,10,
17,11,16,12,15,13,14
```

```
5960 DATA 20,14,15,13,16,12,17,11,18,10,
19,9,1,8,2,7,3,6,4,5
5970 DATA 5,20,6,4,7,3,8,2,9,1,10,19,11,
18,12,17,13,16,14,15
5980 DATA 20,15,16,14,17,13,18,12,19,11,
1,10,2,9,3,8,4,7,5,6
5990 DATA 6,20,7,5,8,4,9,3,10,2,11,1,12,
19,13,18,14,17,15,16
6000 DATA 20,16,17,15,18,14,19,13,1,12,2
,11,3,10,4,9,5,8,6,7
6010 DATA 7,20,8,6,9,5,10,4,11,3,12,2,13
,1,14,19,15,18,16,17
6020 DATA 20,17,18,16,19,15,1,14,2,13,3,
12,4,11,5,10,6,9,7,8
6030 DATA 8,20,9,7,10,6,11,5,12,4,13,3,1
4,2,15,1,16,19,17,18
6040 DATA 20,18,19,17,1,16,2,15,3,14,4,1
3,5,12,6,11,7,10,8,9
6050 DATA 9,20,10,8,11,7,12,6,13,5,14,4,
15,3,16,2,17,1,18,19
6060 DATA 20,19,1,18,2,17,3,16,4,15,5,14
,6,13,7,12,8,11,9,10
6070 DATA 10,20,11,9,12,8,13,7,14,6,15,5
,16,4,17,3,18,2,19,1
6080 REM Data voor ere divisie (D)
          voorbeeld voor 20 clubs
6090 DATA Hannover,Aachen,Hertha,Wattens
.,Saarbru.
6100 DATA Solingen,Kassel,Nuernbe.,Hombu
rg,Buersta.
6110 DATA Offenba.,Ulm,Koeln,Freiburg,Da
rmsta.
6120 DATA Oberhau.,Stuttga.,BW 90,St.Pau
li,Duisburg
6130 REM Data voor de ere divisie (NL)
         voorbeeld voor 18 clubs
6140 DATA Ajax,P.S.V.,Feyen.,Groning.,Sp
arta,Twenthe
6150 DATA Den Bosch,Utrecht,Roda,MVV,Haa
rlem,Volendam,Sittard.
```

```
6160 DATA Exelsior,AZ 67,Go.Eagl.,NAC,Zw
olle
6170 REM Data voor 16 clubs
6180 DATA FC.VVV,Heerenvn.,RKC,Heracl.,G
raafsch.,Den Haag,Cambuur,NEC
6190 DATA Vitesse,WillemII,Veendam,Wagen
,Telstar,SVV,Helmond,RBC,
6200 REM DATA voor 14 clubs
6210 DATA A,B,C,D,E,F,G
6220 DATA H,I,J,K,L,M,N
6230 REM Data voor 12 clubs
6240 DATA A,B,C,D,E,F
6250 DATA G,H,I,J,K,L
6260 REM Data voor 10 clubs
6270 DATA A,B,C,D,E
6280 DATA F,G,H,I,J
6290 REM Data voor 8 clubs
6300 DATA A,B,C,D
6310 DATA E,F,G,H
6320 REM Data voor 6 clubs
6330 DATA A,B,C
6340 DATA D,E,F
6350 REM Data voor 4 clubs
6360 DATA A,B
6370 DATA C,D
61000 REM Omzetten >chr$(127)->negatief
61010 FOR M=2304 TO 3055
61020 A$=BIN$(VPEEK(M))
61030 A$=STRING$(8-LEN(A$),"0")+A$
61040 FOR N=1 TO 8
61050 IF MID$(A$,N,1)="1" THEN MID$(A$,N
,1)="0" ELSE MID$(A$,N,1)="1"
61060 NEXT N
61070 A$="&b"+A$
61080 VPOKE M+768,VAL(A$)
61090 A$=""
61100 NEXT M
61110 RETURN
61120 REM Omzetten in negatief schrift
61130 REM a=0 -> alles
```

```
61140 REM a=1 -> tot aan spatie
61150 REM a=2 -> alleen kleine letters
61160 REM a=3 -> alleen hoofdletters
61170 REM a=4 -> alleen cijfers
61180 FOR PQ=1 TO LEN(A$)
61190 IF A=0 THEN GOTO 61240
61200 AF A=1 AND MID$(A$,PQ,1)=" " THEN
GOTO 61250
61210 IF A=2 AND (MID$(A$,PQ,1)<CHR$(97)
 OR MID$(A$,PQ,1)>CHR$(125)) THEN GOTO 6
1250
61220 IF A=3 AND (MID$(A$,PQ,1)<CHR$(65)
 OR MID$(A$,PQ,1)>CHR$(93)) THEN GOTO 61
250
61230 IF A=4 AND (MID$(A$,PQ,1)>CHR$(57)
 OR MID$(A$,PQ,1)<CHR$(48)) THEN GOTO 61
250
61240 MID$(A$,PQ,1)=CHR$(ASC(MID$(A$,PQ,
1))+96)
61250 NEXT PQ
61260 PRINT A$
61270 RETURN
```

Hollow – het kersenspel

Een spel zoals in een speelhal, dat alle grafische mogelijkheden van uw MSX-computer volledig uitbuit.

Waar gaat het om? Laten we Lars, een 14-jarige gymnasiast uit Munster zelf maar eens aan het woord laten. Hij kreeg het idee voor dit spel en heeft het grootste deel van dit programma in een stil kamertje zelf geprogrammeerd.

Hollow

Op naar de vrolijke kersenjacht. U moet met een vogel zoveel mogelijk kersen opeten. U mag hierbij in geen geval met een andere vogel in botsing komen. Elke keer als u 20 kersen opgegeten heeft, veranderen de andere vogels, bij wijze van bonus, in kersen, die u dan ook weer op kunt eten. Als u 500 punten heeft verworven wordt het spel moeilijker. Als u 700 punten heeft bereikt en niet meer de beschikking heeft over alle levens, krijgt u een extra leven. Een en ander wordt bestuurd met de joystick of het toetsenbord. Omhoog met de vuurknop of de spatiebalk. Naar beneden gaat vanzelf door de zwaartekracht!

Het is voorwaar niet gemakkelijk, zelfs niet voor Lars, onze jeugdige programmeur, om meer dan 500 punten te scoren. Hoe komt dat? Vanaf 500 punten storten de vogels hoe langer hoe ongecontroleerder in steile vluchten op en neer. Vaak precies op de plek waar u uw kersen wilde eten. Maar probeer het rustig over u heen te laten komen. Lars heeft geen tijdslimiet in het programma 'Hollow' ingebouwd.

Voor wat de programmatuur betreft, zullen u in eerste instantie de vele DATA-regels opvallen. Hier zijn de vogels als sprites in verschillende posities vastgelegd. Maar ook de kersen die u opeten moet/wilt. Als u er goed naar kijkt, zult u opmerken dat de kersen niet alleen met sprites maar op andere plaatsen ook met een 'DRAW'-commando worden opgebouwd. Misschien herinnert u zich nog dat de andere vogels veranderen in kersen (sprites). Op andere momenten hoeft u als vogel alleen maar naar de stilstaande kersen toe te vliegen (stilstaand als 'DRAW'-commando).

Botst u tegen de vogels (sprites) aan, dan treedt het commando 'ON SPRITE GOSUB' in werking en u stevent af op een snelle duikvlucht ofwel speldood. Het geluid is een beetje magertjes, maar anders zou de snelheid op het beeld te langzaam worden, aldus Lars!? Indien Lars destijds al de beschikking zou hebben gehad over de 'geluidseditor', dan zouden ten minste fladdergeluiden zonder onderbrekingen tijdens het spel tot de mogelijkheden hebben behoord.

(Eenmaal met 'SOUND' geprogrammeerde zelf gemaakte geluiden zouden continu hoorbaar zijn zonder onderbrekingen.) Nu is het echter zo, dat er alleen een begin- en eindmelodie weerklinken en wat geluiden bij bijzondere acties zoals het kersen eten, de speldood e.d.

Maar nu hebben we genoeg gezegd. Veel plezier met het programmeren en het spelen toegewenst, namens Lars en de auteur.

```
10 REM Spel Hollow
20 REM MSX Verzamelde programma's
30 REM Copyright 1984 DATA BECKER &
       Rainer Lueers & Lars Lewejohann
40 REM Kersen stilstaand
50 B$="f211h111g1d1f312f1b14e112e2"
60 STOP ON
70 ON STOP GOSUB 4460
80 COLOR 10,1,1
90 SCREEN 2,2,0
100 OPEN"grp:" FOR OUTPUT AS #1
110 DRAW "bm 100,85"
120 PRINT #1,"HOLLOW"
130 GOSUB 4910
140 DEFINT A-Z
150 REM Sprite-definitie 8*8
160 FOR A=9 TO 11
170 S$=""
180 FOR B=1 TO 8
190 READ A$
200 S$=S$+CHR$(VAL("&b"+A$))
210 NEXT B
220 SPRITE$(A)=S$
230 NEXT A
240 REM Rotspunten onder
250 DATA 00000000
260 DATA 00000000
270 DATA 00000000
280 DATA 00001000
290 DATA 00011000
300 DATA 00011100
310 DATA 01111100
320 DATA 11111111
330 '
340 REM Rotspunten boven
350 DATA 11111111
360 DATA 01111100
370 DATA 00011100
380 DATA 00011000
390 DATA 00001000
```

```
400 DATA 00000000
410 DATA 00000000
420 DATA 00000000
430 '
440 REM Kersen bewegend
450 DATA 00000100
460 DATA 00001110
470 DATA 00010011
480 DATA 00010000
490 DATA 00101000
500 DATA 01000100
510 DATA 11101110
520 DATA 01000100
530 REM Spritedefinitie 32*32
540 FOR A=0 TO 8
550 S$=SPACE$(64)
560 FOR I%=1 TO 16
570 READ A$
580 MID$(S$,I%,1)=CHR$(VAL("&b"+LEFT$(A$
,8)))
590 MID$(S$,I%+16,1)=CHR$(VAL("&b"+RIGHT
$(A$,8)))
600 NEXT I%
610 SPRITE$(A)=S$
620 NEXT A
630 '
640 REM Vogel lopend rechts
650 DATA 0000000000000000
660 DATA 0000000000000000
670 DATA 0000000000000000
680 DATA 0000000000000000
690 DATA 1000000000001100
700 DATA 1101111110001110
710 DATA 1111111111011000
720 DATA 0111110011110000
730 DATA 0111001111100000
740 DATA 0011111111000000
750 DATA 0001111110000000
760 DATA 0000010010000000
770 DATA 0000100001000000
```

```
780 DATA 0111000000100000
790 DATA 0100000000110000
800 DATA 0000000000000000
810 '
820 DATA 0000000000000000
830 DATA 0000000000000000
840 DATA 0000000000000000
850 DATA 0000000000000110
860 DATA 1000000000000111
870 DATA 1101111110001100
880 DATA 1111111111011000
890 DATA 0111110011110000
900 DATA 0111001111100000
910 DATA 0011111111000000
920 DATA 0001111111000000
930 DATA 0000100010000000
940 DATA 0000010100000000
950 DATA 0000001000000000
960 DATA 0000110100000000
970 DATA 0000100011000000
980 '
990 REM Vogel vliegend rechts
1000 DATA 0000000000000000
1010 DATA 0000001000000000
1020 DATA 0010001100000000
1030 DATA 0011001010000000
1040 DATA 1001101101000000
1050 DATA 1101010101001100
1060 DATA 1111101111011110
1070 DATA 0111111111110000
1080 DATA 0111111111100000
1090 DATA 0011111111000000
1100 DATA 0001111111000000
1110 DATA 0000010010000000
1120 DATA 0000100100000000
1130 DATA 0001001000000000
1140 DATA 0010010000000000
1150 DATA 0001001000000000
1160 '
1170 DATA 0000000000000000
```

```
1180 DATA 0000000000000000
1190 DATA 0000000000000000
1200 DATA 0000000000000110
1210 DATA 1000000000000111
1220 DATA 1101111110001100
1230 DATA 1111111111011000
1240 DATA 0111110111110000
1250 DATA 0111101111100000
1260 DATA 0011011111000000
1270 DATA 0001111111000000
1280 DATA 0000010010000000
1290 DATA 0000100100000000
1300 DATA 0001001000000000
1310 DATA 0010010000000000
1320 DATA 0001001000000000
1330 '
1340 REM Vogel lopend links
1350 DATA 0000000000000000
1360 DATA 0000000000000000
1370 DATA 0000000000000000
1380 DATA 0000000000000000
1390 DATA 0000000000000001
1400 DATA 0011000111111011
1410 DATA 0111101111111111
1420 DATA 0000111100111110
1430 DATA 0000011111001110
1440 DATA 0000001111111100
1450 DATA 0000000111111000
1460 DATA 0000000100100000
1470 DATA 0000001000010000
1480 DATA 0000010000001110
1490 DATA 0001100000000010
1500 DATA 0000000000000000
1510 '
1520 DATA 0000000000000000
1530 DATA 0000000000000000
1540 DATA 0000000000000000
1550 DATA 0110000000000000
1560 DATA 1110000000000001
1570 DATA 0011000111111011
```

```
1580 DATA 0001101111111111
1590 DATA 0000111100111110
1600 DATA 0000011111001110
1610 DATA 0000001111111100
1620 DATA 0000000111111000
1630 DATA 0000000100010000
1640 DATA 0000000010100000
1650 DATA 0000000001000000
1660 DATA 0000000010110000
1670 DATA 0000001100010000
1680 '
1690 REM Vogel vliegend links
1700 DATA 0000000000000000
1710 DATA 0000000001000000
1720 DATA 0000000011000100
1730 DATA 0000000101001100
1740 DATA 0000001011011001
1750 DATA 0011001010101011
1760 DATA 0111101111011111
1770 DATA 0000111111111110
1780 DATA 0000011111111110
1790 DATA 0000001111111100
1800 DATA 0000000111111000
1810 DATA 0000000100100000
1820 DATA 0000000010010000
1830 DATA 0000000001001000
1840 DATA 0000000000100100
1850 DATA 0000000001001000
1860 '
1870 DATA 0000000000000000
1880 DATA 0000000000000000
1890 DATA 0000000000000000
1900 DATA 0110000000000000
1910 DATA 1110000000000001
1920 DATA 0011000111111011
1930 DATA 0001101111111111
1940 DATA 0000111101111110
1950 DATA 0000011110111110
1960 DATA 0000001111011100
1970 DATA 0000000111111000
```

```
1980 DATA 0000000100100000
1990 DATA 0000000010010000
2000 DATA 0000000001001000
2010 DATA 0000000000100100
2020 DATA 0000000001001000
2030 '
2040 REM Vogel dood
2050 DATA 0000000000000000
2060 DATA 0000000000000000
2070 DATA 0000000000000000
2080 DATA 0000000000000000
2090 DATA 0000000000000000
2100 DATA 0000000000000000
2110 DATA 1000000000001100
2120 DATA 1101111100010010
2130 DATA 1111100111001100
2140 DATA 0110011111000000
2150 DATA 0011111110100000
2160 DATA 0100000010011100
2170 DATA 1011111100001110
2180 DATA 0100000000000000
2190 DATA 0000000000000000
2200 DATA 0000000000000000
2210 '
2220 CLS
2230 DRAW "bm 0,0"
2240 COLOR 15
2250 PRINT #1,TAB(12);"HOLLOW"
2260 COLOR 3
2270 PRINT #1,TAB(12);"======"
2280 COLOR 10
2290 PRINT #1,
2300 PRINT #1,
2310 PRINT #1,
2320 PRINT #1,"      Wilt u de":PRINT #1,
"      Instructies lezen?"
2330 COLOR 10
2340 PRINT #1,
2350 PRINT #1,"               (J/N)"
2360 C$=INKEY$
```

```
2370 IF C$="J" OR C$="j" THEN GOTO 4960
ELSE IF C$="N" OR C$="n" THEN GOTO 2380
ELSE GOTO 2360
2380 CLS
2390 DRAW"bm 0,0
2400 COLOR 15
2410 PRINT #1,TAB(10);"HOLLOW"
2420 COLOR 3
2430 PRINT #1,TAB(10);"======"
2440 COLOR 10
2450 PRINT #1,
2460 PRINT #1,
2470 PRINT #1,"    Wilt u met "
2480 COLOR 15
2490 PRINT #1,"    de joystick"
2500 COLOR 10
2510 PRINT #1,"    of";
2520 PRINT #1," met het "
2530 COLOR 15
2540 PRINT #1,"    toetsenbord"
2550 COLOR 10
2560 PRINT #1,"      spelen?"
2570 PRINT #1,
2580 PRINT #1,TAB(10);"(J/T)"
2590 C$=INKEY$
2600 IF C$="T" OR C$="t" THEN SZ=0 ELSE
IF C$="J" OR C$="j" THEN SZ=1 ELSE GOTO
2590
2610 IF MU=1 THEN GOSUB 4910
2620 REM Het vastleggen van de
        variabelen m.b.t het
        aantal levens e.d. die niet
        meer vehoogd mogen worden
2630 MS=0
2640 SM=0
2650 S=SZ
2660 PU=0
2670 PR(1)=0
2680 PR(2)=0
2690 PR(3)=0
```

```
2700 EN=4
2710 TI=0
2720 CLS
2730 REM Grottenwand
2740 LINE (0,148)-(256,192),2,BF
2750 LINE (0,40)-(256,0),2,BF
2760 COLOR 15
2770 REM Levens aangeven
2780 DRAW "c15bm16,3d4r2br2u4bm22,3d3f1e
1u3br2d4r2l2u2r1l1u2r2br4l2d2r2d2l2"
2790 PSET (20,1),15
2800 PSET (35,1),15
2810 PSET (35,6),15
2820 REM Score aangeven
2830 DRAW "c15bm125,7r2u2l2u2r2br2r2l2d4
r2br2r2u4l2d4r2br2u4r2d1g1f1d1br2u4r2l2d
2r1l1d2r2"
2840 PSET (146,4),15
2850 PSET (146,6),15
2860 DRAW "bm 118,10":PRINT #1,PU
2870 REM High-score aangeven
2880 DRAW "c15bm220,7u4d2r2u2d4br2u4d2br
2r2bd2br2r2u2l2u2r2br2r2l2d4r2br2r2u4l2d
4r2br2u4r2d1g1f1d1br2u4r2l2d2r1l1d2r2"
2890 PSET (250,4),15
2900 PSET (250,6),15
2910 PSET (250,1),15
2920 DRAW "bm 213,10":PRINT #1,HC
2930 REM variabelen die voor het
         hoofdprogramma gebruikt worden
2940 FR=110
2950 Z=182
2960 ZR=110
2970 G=0
2980 F=82
2990 H=0
3000 K=0
3010 RZ=3
3020 RF=-4
3030 RG=2
```

```
3040 R=12
3050 W=1
3060 X=100
3070 Y=133
3080 L=0
3090 GS=5
3100 FS=5
3110 ZS=5
3120 GR=110
3130 I=0
3140 J=0
3150 CO=10
3160 KI=0
3170 KK=0
3180 TR=0
3190 UT=122
3200 UH=48
3210 IF MS=1 THEN PLAY "l64cr64er64dr64f
r64ar64f":MS=2
3220 IF SM=1 THEN PLAY "l64cr64er64dr64f
r64ar64f":SM=2:GOSUB 4860
3230 IF PU>490 THEN RZ=8:RF=11:RG=10:UT+
115:UH=55
3240 REM Sprites in uitgangspositie
         plaatsen
3250 PUT SPRITE 0,(X,Y),10,0
3260 PUT SPRITE 1,(K,139),2,9
3270 PUT SPRITE 2,(H+90,139),2,9
3280 PUT SPRITE 3,(G,110+GR),5,GS
3290 PUT SPRITE 4,(F+82,110+FR),4,FS
3300 PUT SPRITE 7,(Z+164,110+ZR),3,ZS
3310 PUT SPRITE 5,(I+30,40),2,10
3320 PUT SPRITE 6,(J+180,40),2,10
3330 PUT SPRITE 8,(12,4),11,PR(1)
3340 PUT SPRITE 9,(30,4),11,PR(2)
3350 PUT SPRITE 10,(48,4),11,PR(3)
3360 REM De stilstaande kersen
         verwijderen indien aanwezig
3370 SPRITE ON
3380 ON SPRITE GOSUB 4290
```

```
3390 PSET (KX,KY),1
3400 DRAW B$
3410 REM Nieuwe kersen plaatsen
        (stiltaande)
3420 TIME=RND(-TIME)*65535!
3430 KX=20+RND(-TIME)*255
3440 KY=44+RND(-TIME)*90
3450 PSET (KX,KY),0
3460 DRAW "c8"+B$
3470 REM Hoofdprogramma
3480 A%=STICK(S)
3490 REM Het kersen eten
3500 IF TI<>-1 AND KX-2>X AND KX-2<X+16
AND KY+4>Y AND KY+4<Y+16 THEN GOSUB 4580
3510 IF TI=-1 THEN PSET (KX,KY),1:DRAW B
$
3520 IF TI=20 THEN GOSUB 4510
3530 REM Bonus-kersen eten
3540 IF GR=209 AND FR=209 AND ZR=209 THE
N PLAY "som2000o4l16br7br5g":SPRITE ON:T
I=0:ET=0:GOTO 2940
3550 IF KI=1 AND GR+4>=Y AND GR+4<=Y+16
AND G+4>=X AND G+4<=X+16 THEN RG=0:GR=20
9:KK=1:GOSUB 4580
3560 IF KI=1 AND FR+4>=Y AND FR+4<=Y+16
AND F+4>=X AND F+4<=X+16 THEN RF=0:FR=20
9:KK=1:GOSUB 4580
3570 IF KI=1 AND ZR+4>=Y AND ZR+4<=Y+16
AND Z+4>=X AND Z+4<=X+16 THEN RZ=0:ZR=20
9:KK=1:GOSUB 4580
3580 REM Rotspunten plaatsen
3590 IF Y=131 AND OT=1 THEN GOTO 4290
3600 IF KI<>1 THEN K=K-7:PUT SPRITE 1,(K
,139),2,9:IF K<0 THEN K=256
3610 IF KI=1 THEN PUT SPRITE 1,(0,209),2
,9
3620 IF KI<>1 THEN H=H-7:PUT SPRITE 2,(H
+90,139),2,9:IF H<0 THEN H=256
3630 IF KI=1 THEN PUT SPRITE 2,(0,209),2
,9
```

```
364Ø IF KI<>1 THEN CO(1)=5:GS=GS+1 ELSE
GS=11:CO(1)=6
365Ø REM De vogels bewegen
366Ø GR=GR+RG
367Ø G=G-11
368Ø PUT SPRITE 3,(G,GR),CO(1),GS
369Ø IF G<Ø THEN G=256
37ØØ IF KI<>1 THEN IF GS>6 THEN GS=5
371Ø IF GR>UT THEN RG=-RG ELSE IF GR<UH
THEN RG=-RG
372Ø IF KI<>1 THEN CO(2)=4:FS=FS+1 ELSE
FS=11:CO(2)=9
373Ø FR=FR-RF
374Ø F=F-11
375Ø PUT SPRITE 4,(F,FR),CO(2),FS
376Ø IF F<Ø THEN F=256
377Ø IF KI<>1 THEN IF FS>6 THEN FS=5
378Ø IF FR>UT THEN RF=-RF ELSE IF FR<UH
THEN RF=-RF
379Ø IF KI<>1 THEN ZS=ZS+1 ELSE ZS=11
38ØØ ZR=ZR-RZ
381Ø Z=Z-11
382Ø PUT SPRITE 7,(Z,ZR),CO(1),ZS
383Ø IF Z<Ø THEN Z=256
384Ø IF KI<>1 THEN IF ZS>6 THEN ZS=5
385Ø IF ZR>UT THEN RZ=-RZ ELSE IF ZR<UH
THEN RZ=-RZ
386Ø IF KI<>1 THEN I=I-7:PUT SPRITE 5,(I
+3Ø,4Ø),2,1Ø:IFI=<Ø THEN I=256
387Ø IF KI=1 THEN PUT SPRITE 5,(Ø,2Ø9),2
,1Ø
388Ø IF KI<>1 THEN J=J-7:PUT SPRITE 6,(J
+18Ø,4Ø),2,1Ø:IF J<Ø THEN J=256
389Ø IF KI=1 THEN PUT SPRITE 6,(Ø,2Ø9),2
,1Ø
39ØØ REM Besturing van de hoogte
         van de vogels
391Ø IF STRIG(S) THEN RU=-6 ELSE IF S=2
THEN RU=16 ELSE RU=9
392Ø IF STRIG(S) AND A%=Ø THEN Y=Y-2:GOS
UB 418Ø
```

```
3930 IF Y<178 AND A%=0 AND STRIG(S)=0 TH
EN Y=Y+2:GOSUB 4180
3940 REM Vogel met kop naar rechts
3950 IF A%=0 THEN W=1
3960 REM Bewegen v.d. vogels (Joystick
         bekijken) en loopritme
3970 ON A% GOSUB 3980,3990,4020,4050,408
0,4090,4120,4150:GOTO 3480 ELSE GOTO 393
0
3980 GOTO 4180
3990 W=1
4000 X=X+R
4010 GOTO 4180
4020 W=1
4030 X=X+R
4040 GOTO 4180
4050 W=1
4060 X=X+R
4070 GOTO 4180
4080 GOTO 4180
4090 W=2
4100 X=X-R
4110 GOTO 4180
4120 W=2
4130 X=X-R
4140 GOTO 4180
4150 W=2
4160 X=X-R
4170 GOTO 4180
4180 IF W=1 AND Y>=131 THEN SP=SP+1:IF S
P>1 THEN SP=0
4190 IF W=2 AND Y>=131 THEN SP=SP+1:IF S
P>5 THEN SP=4
4200 IF W=1 AND Y<131 THEN SP=SP+1:IF SP
>3 THEN SP=2
4210 IF W=2 AND Y<131 THEN SP=SP+1:IF SP
>7 THEN SP=6
4220 Y=Y+RU
4230 IF Y>131 THEN Y=131
```

```
4240 IF X>240 THEN X=240
4250 IF Y<40 THEN Y=40
4260 IF X<8 THEN X=8
4270 PUT SPRITE 0,(X,Y),CO,SP
4280 RETURN
4290 SPRITE OFF
4300 REM levenslust bezien dood of niet
4310 S=2
4320 IF Y=131 THEN PUT SPRITE 0,(X,134),
14,8:PLAY "l24som9000o1cl5o5som2000":FOR
 A=1 TO 2000:NEXT:S=SZ:OT=0:GOTO 4330 EL
SE S=2:OT=1:CO=14:PLAY "l24som9000o2a":G
OTO 3480
4330 EN=EN-1
4340 GOSUB 4840
4350 IF EN<1 THEN DRAW "bm 100,96":PRINT
 #1,"Game over":IF PU>HC THEN HC=PU:COLO
R 2:DRAW "bm 213,10":PRINT #1,STRING$(6,
CHR$(200)):DRAW "bm 213,10":COLOR 15:PRI
NT #1,HC:GOTO 4370
4360 IF EN>=1 THEN GOTO 2940
4370 DRAW "bm 30,170"
4380 PRINT #1,"Druk een toets a.u.b."
4390 A$=INKEY$
4400 IF A$<>"" THEN GOTO 4390
4410 A$=INKEY$
4420 IF A$="" THEN GOTO 4410
4430 MU=1
4440 GOTO 2610
4450 REM Einde van het spel en terug
         keren met een schoon SCREEN 0
4460 SCREEN 0,,1
4470 COLOR 15,4,15
4480 KEY OFF
4490 END
4500 REM Vogels in kersen veranderen
4510 SPRITE OFF
4520 ET=1
4530 TI=-1
4540 KI=1
```

```
4550 PLAY "l5o5fco4bag"
4560 RETURN
4570 REM De punten
4580 IF KK=1 THEN TI=TI-1:TR=1
4590 PLAY "l5som2000o5go4"
4600 TI=TI+1
4610 PU=PU+10
4620 DRAW "bm 118,10"
4630 COLOR 2
4640 PRINT #1,STRING$(7,CHR$(200))
4650 COLOR 15
4660 DRAW "bm 118,10"
4670 PRINT #1,PU
4680 REM Bonus of niet
4690 IF PU=500 AND MS=0 THEN MS=1:GOTO 2
940
4700 REM Bonus of niet. Ja, dan wordt
         het spel moeilijker
4710 IF PU=700 AND SM=0 THEN SM=1:GOTO 2
940
4720 IF TR=1 THEN RETURN
4730 REM De stilstaande kersen plaatsen
4740 LINE (KX-8,KY-1)-(KX+5,KY+11),1,BF
4750 TIME=RND(-TIME)*65535!
4760 KX=20+RND(-TIME)*255
4770 KY=44+RND(-TIME)*90
4780 PSET (KX,KY),1
4790 DRAW "c8"+B$
4800 KK=0
4810 TR=0
4920 RETURN
4830 REM Aantal levens tegoed afbeelden
4840 IF EN=3 THEN PR(1)=8 ELSE IF EN=2 T
HEN PR(2)=8 ELSE IF EN=1 THEN PR(3)=8
4850 RETURN
4860 IF EN=4 THEN RETURN
4870 IF EN=3 THEN PR(1)=0 ELSE IF EN=2 T
HEN PR(2)=0 ELSE IF EN=1 THEN PR(3)=0
4880 EN=EN+1
4890 RETURN
```

```
4900 REM Openingsmelodie
4910 PLAY "t255som2000o4l16br7br5gr3br7b
r5gr3"
4920 PLAY "l5o5fco4bagr50"
4930 PLAY "l64ar7ar5fr3ar7ar5fr3"
4940 RETURN
4950 REM Uitleg
4960 SCREEN 0:WIDTH 40
4970 LOCATE 0,0
4980 PRINT TAB(13) "HOLLOW"
4990 PRINT TAB(13) "======="
5000 PRINT
5010 PRINT "Op naar de vrolijke kersenja
cht. U moet er met een vogel zoveel moge
lijk          opeten."
5020 PRINT "U mag daarbij niet in botsin
g komen met de andere vogels."
5030 PRINT "Telkens als u 20 kersen opge
geten heeft veranderen de vogels als bon
us ook in    kersen. Die u dan weer eten
kunt. Als u 500 punten heeft wordt het s
pel           moeilijker."
5040 PRINT "Als u 700 punten heeft en ni
et meer       alle levens dan krijgt u een
 extra        leven."
5050 PRINT "Sturen doet u met de joystic
k of het      toetsenbord. U zoekt het hog
erop met de vuurknop of de spatiebalk. N
aar beneden"
5060 PRINT "gaat vanzelf door de zwaarte
kracht."
5070 PRINT:PRINT "Druk een toets a.u.b..
.."
5080 C$=INKEY$:IFC$=""THEN GOTO 5080 ELS
E RUN
```

Reactietijd

Om mensen te verrassen, maar misschien ook wel om mensen die zich tot dusver negatief opstelden tegenover het hele computergebeuren, wat positiever te stemmen voor wat betreft deze wondermachine, wordt dit programma als demonstratie ten zeerste aanbevolen.

De essentie van de programmatuur is: er moet zo snel als enigszins mogelijk een toets ingedrukt worden als de computer hier om vraagt. Hierna volgt een waardering en wordt u andermaal uitgenodigd een nieuw spel met maximaal vier deelnemers te beginnen.

De clou van het programma schuilt voornamelijk in de wijze waarop de nietsvermoedende mens achter de computer behandeld wordt. Zoals reeds eerder in dit boek werd opgemerkt: computers zijn allemaal heel erg dom. Ze worden pas door het programmeren door mensen slim gemaakt. En dat is nu precies wat de leek als gebruiker van dit programma duidelijk zal worden.

Worden er bij het begin van het programma geen namen ingevoerd, dan wordt uw MSX-computer ofwel het programma, langzaam maar zeker chagrijnig en begint ten slotte te dreigen met: men maakt de computer door fouten kapot! Gaat het wel allemaal volgens de regels dan antwoordt uw MSX-computer met een sympathiek 'dank u'.

Tijdens het spel let uw MSX-computer erop dat er niet vals gespeeld wordt. Als dit wel het geval is, heeft dit een berisping en een tijdstraf tot gevolg.

De gemeten tijd wordt tot op een honderdste van een seconde nauwkeurig aangegeven. Bovendien geeft uw MSX-computer er nog zijn commentaar op, zoals: ('SUPER' of 'heel zwak').

Programmatechnisch gezien een makkie. Maar het wordt door de medemens wellicht gezien als een goed argument voor de zeer verstandige aankoop (!) van uw MSX-computer. Misschien maakt u van enkelen onder hen nog wel een computerfreak. En dan is de vreugde en het enthousiasme wellicht dubbel zo groot als voorheen. Nog een toepassingsmogelijkheid van dit programma: meet uw reactietijd maar eens onder verschillende gemoedstoestanden!

```
Reinier had
.62 seconden nodig!

Ellen had
.72 seconden nodig!

Nicole had
.3 seconden nodig!

Marco had
.48 seconden nodig!

Nog een spelletje ( /N) ?
```

```
10 REM Reaktie
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                         Rainer Lueers
40 REM Voorbereidingen treffen
50 COLOR 15,1,1:SCREEN 0:WIDTH 40:KEY OF
F
60 INPUT "Hoeveel deelnemers (1 t/m 4) "
;F$
70 IF VAL(F$)<1 OR VAL(F$)>4 THEN GOTO 6
0
80 FOR N=1 TO VAL (F$)
90 PRINT:PRINT  "Uw naam a.u.b. speler";
N;:INPUT F1$(N)
100 REM reaktie op verkeerde invoer
110 IF F1$(N)="" AND ME=0 THEN PRINT:PRI
NT "Wees toch niet zo bang!
    Alles heeft toch een naam. Voorzover
 ik als domme computer dat weet tenminst
e.  Geldt dat toch ook voor mensen en
    dieren!":ME=1:GOTO 90
120 IF F1$(N)="" AND ME=1 THEN PRINT:PRI
NT "Alle mensen die mensen toch!
    Enfin laten we eens rustig aan begin
nen misschien gaat het dan! Mijn naam is
    Standaard. Nee! Ik kom niet van Mars
.   Mijn voornaam is MSX. Grappig niet?"
130 IF F1$(N)="" AND ME=1 THEN PRINT "Je
 mag mij rustig 'homecomputer' noemen ma
ar alleen als je mij je voornaam zegt.No
u komt er nog wat van?":ME=2:GOTO 90
140 IF F1$(N)="" AND ME=2 THEN PRINT:PRI
NT "Nog een keer verkeerde invoer en de
    computer is kapot.. Ik mag toch wel
    hopen dat.........";:ME=3:GOTO 90
150 IF F1$(N)="" AND ME=3 THEN FOR N=1 T
O 10000:PRINT CHR$(PEEK(N));:NEXT
160 ME=0
170 NEXT N
180 CLS:LOCATE 16,11:PRINT "Dank u"
```

```
190 FOR N=1 TO 2000:NEXT N
200 CLS:FOR N=1 TO VAL(F$)
210 PRINT F1$(N);"!!!!":PRINT:PRINT:PRIN
T "Jouw beurt!":PRINT "Druk op een van m
ijn toetsen, als je    gemerkt hebt, dat
 de beeldscherm-        kleur veranderd i
s."
220 REM berekening van de tijdsduur
230 FOR M=1 TO 2000:NEXT M:A=1000*RND(TI
ME):FOR M=1 TO A:A$=INKEY$:IF A$="" THEN
 NEXT M:GOSUB 300 ELSE CLS:PRINT "bedrie
ger!!!!":PRINT "Je krijgt 5 sekonden":PR
INT "als straf.":F(N)=5:FOR M=1 TO 2000:
NEXT M:CLS
240 REM eindresulaat weergeven
250 NEXT N:CLS:FOR N=1 TO VAL(F$):PRINT
F1$(N);" had":PRINT F(N);"seconden nodig
!":PRINT:NEXT N:PRINT:PRINT:INPUT "Nog e
en spelletje ( /N) ";Z$:IF LEFT$(Z$,1)="
N" OR LEFT$(Z$,1)="n" THEN Z$="N"
260 IF Z$="N" THEN END
270 PRINT:INPUT "Met dezelfde spelers (J
/ ) ";Z$:IF LEFT$(Z$,1)="j" OR LEFT$(Z$,
1)="J" THEN Z$="J"
280 IF Z$="J" THEN GOTO 200 ELSE RUN
290 REM Routine voor tijdcontrole
300 CLS:COLOR 1,15:B=TIME
310 A$=INKEY$:IF A$="" THEN GOTO 310
320 C=TIME:COLOR 15,1:CLS:F(N)=(C-B)/50:
PRINT F1$(N):PRINT "Je had";F(N);"second
en nodig!"
330 PRINT
340 REM Waardering
350 IF F(N)<.3 THEN PRINT "S U P E R":GO
TO 390
360 IF F(N)<.35 THEN PRINT "Heel goed":G
OTO 390
370 IF F(N)<.4 THEN PRINT "Zwakjes":GOTO
 390
380 PRINT "Zeer zeer zwak!"
390 FOR M=1 TO 2000:NEXT M:CLS:RETURN
```

Codebreker

Dit spel bestaat reeds in vele uitvoeringen. Het bekendste is het wel onder de naam 'mastermind'.

We hebben het computerprogramma echter omgedoopt in 'codebreker', omdat het zich van het genoemde voorbeeld duidelijk onderscheidt.

Om te beginnen kunt u kiezen uit hoeveel kleuren = getallen de te raden volgorde samengesteld moet worden. Er staan u maximaal acht kleuren = getallen ter beschikking.

Aansluitend wordt u een tweede keuze geboden. Uit hoeveel rijen moet de volgorde bestaan? Het is weliswaar mogelijk meer kleuren dan rijen te benoemen. Daarentegen is in het omgekeerde niet voorzien en dit wordt dan ook genegeerd. Dit heeft de volgende reden. De speler moet van het begin af aan weten met welke strategie hij te maken heeft. Hierdoor zal het duidelijk zijn dat na de keuze van acht kleuren en acht plaatsen ook daadwerkelijk acht kleuren worden gebruikt. Dubbele of driedubbele zijn hiermee uitgesloten.

Op deze basis moet tijdens het spel dan ook te werk worden gegaan. Het heeft geen enkele zin door het invoeren van de getallencombinatie '11112222' te proberen uit te vinden of de '1' of de '2' wellicht meerdere malen in een rij voorkomen.

Op het beeldscherm wordt aangegeven hoeveel kleuren juist en hoeveel kleuren ook nog op de juiste plaats ingevoerd werden. Voor de beginnende 'codebreker' voorwaar een moeilijke opgave. Maar oefening baart kunst. Om het overzicht niet te verliezen, wordt het beeldscherm na iedere invoer omhooggeschoven. De laatste vier à vijf opgaven blijven zichtbaar. Bovendien worden alle resultaten en getallen bewaard. Men kan de complete invoer/resultaten door het indrukken van de (ENTER)-toets opnieuw op het beeldscherm brengen. Nu heb ik genoeg uitgelegd. Veel plezier!

```
Een getal met 8 cijfers ? 12345678
Kleuren op de verkeerde plaats : 8
Kleuren op de juiste plaats : 0
Een getal met 8 cijfers ? 23456781
Kleuren op de verkeerde plaats : 6
Kleuren op de juiste plaats : 2
Een getal met 8 cijfers ? 23876541
Kleuren op de verkeerde plaats : 7
Kleuren op de juiste plaats : 1
Een getal met 8 cijfers ? 15234678
Kleuren op de verkeerde plaats : 7
Kleuren op de juiste plaats : 1
Een getal met 8 cijfers ? 654321

Verkeerde invoer!

Een getal met 8 cijfers ?
```

```
10 REM Codebreker
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                         Rainer Lueers
40 REM Het dimensioneren en de invoer
       van de voorwaarden
50 DIM ME$(100),SC(100),WE(100)
60 SCREEN 0:COLOR 15,1:KEY OFF:WIDTH 40
70 INPUT "Hoeveel kleuren (1 t/m 8) ";F$
80 IF VAL(F$)<1 OR VAL(F$)>8 THEN GOSUB
470:GOTO 70
90 INPUT "Aantal plaatsen (1 t/m 8) ";F1
$
100 IF VAL(F1$)<1 OR VAL(F1$)>8 THEN GOS
UB 470:GOTO 90
110 IF VAL(F1$)>VAL(F$) THEN GOSUB 470:G
OTO 90
120 REM Het kiezen van de getallen
        m.b.v. de RND=toevalsfunctie
130 FOR N=1 TO VAL(F1$)
140 A=INT(10*RND(TIME))
150 IF A<1 OR A>VAL(F$) THEN GOTO 140
160 REM Alle gekozen getallen moeten
        verschilend zijn
170 FOR M=1 TO N-1
180 A(M)=VAL(MID$(A$,M,1))
190 NEXT M
200 FOR M=1 TO N-1
210 IF A=A(M) THEN GOTO 140 ELSE NEXT M
220 A$=A$+RIGHT$(STR$(A),1)
230 NEXT N
240 FOR N=1 TO VAL(F1$)
250 A(N)=VAL(MID$(A$,N,1))
260 NEXT N
270 CLS
280 REM Invoerroutine
290 F3$="":PRINT "Een getal met";VAL(F1$
);"cijfers ";:INPUT F3$
```

```
300 REM Het weergeven van alle getallen
        welke tot dusver werden
        ingevoerd indien geen invoer
        volgt
310 IF F3$="" THEN CLS:FOR Z1=1 TO Z:PRI
NT "beurt"Z1;TAB(7);ME$(Z1);"     OK:";SC
(Z1);"  not OK;";WE(Z1):NEXT Z1:PRINT:GO
TO 290 ELSE IF LEN(F3$)<>VAL(F1$) THEN G
OSUB 470:GOTO 290
320 FOR N=1 TO VAL(F1$)
330 B(N)=VAL(MID$(F3$,N,1))
340 IF A(N)=B(N) THEN SC=SC+1:C(N)=-1
350 NEXT N
360 FOR M=1 TO VAL(F1$)
370 FOR N=1 TO VAL(F1$)
380 IF A(N)=B(M) AND N<>M THEN WE=WE+1:G
OTO 410
390 NEXT N
400 REM Weergave-routine en aansluitend
        testen op juistheid
410 NEXT M:PRINT "Kleuren op de verkeerd
e plaats :";WE
420 Z=Z+1:ME$(Z)=F3$:WE(Z)=WE:SC(Z)=SC:W
E=0
430 PRINT "Kleuren op de juiste plaats :
";SC
440 IF SC=VAL(F1$) THEN PRINT:PRINT CHR$
(200) "Helemaal goed in";Z;"beurten" CHR
$(200):PRINT:INPUT "Nog een keer (J/ ) "
;F$:IF LEFT$(F$,1)="j" OR LEFT$(F$,1)="J
" THEN F$="J" ELSE GOTO 450 ELSE SC=0:GO
TO 290
450 IF F$="J" THEN RUN ELSE END
460 GOTO 290
470 PRINT:PRINT CHR$(200) "Verkeerde inv
oer!" CHR$(200):PRINT:RETURN
```

Talstelsels omrekenen

Uw MSX-computer is niet alleen in staat om getallen in het ons vertrouwde decimale stelsel af te beelden, hij heeft ook speciale commando's om getallen om te rekenen naar andere talstelsels. Zo betekent '&H' voorafgaande aan een getal: hexadecimaal, '&B' betekent binair ofwel tweetallig.

Op deze manier kunnen we 'PRINT &HFF' invoeren en krijgen we dit hexadecimale getal omgerekend in een decimaal getal, nl. '255', als uitkomst op het beeldscherm.

Toetsen we 'PRINT &B111' in, dan krijgen we als uitkomst dit binaire getal omgerekend in een decimaal getal, n.l. '7'.

Moeilijker wordt het al als we direct een hexadecimaal getal willen omrekenen naar een binair getal. Het wordt echter helemaal onmogelijk als we met een ander gangbaar talstelsel dan hexadecimaal, decimaal of binair willen gaan werken. Bij voorbeeld octaal (8-tallig).

Teneinde de talstelsels beter te leren kennen, is dit programma ontstaan. U kunt door het voorvoegen van de betreffende hoofdletter ('H' voor hexadecimaal, 'B' voor binair en 'D' voor decimaal) ieder opgegeven getal laten omrekenen in elk van deze drie talstelsels.

Bovendien kunt u door het indrukken van de (ENTER)-toets uw decimale getal omrekenen in een talstelsel tussen de 2 en 9. Hierbij gebeurt iets interessants en tegelijkertijd leerzaams. Het getal van het gewenste talstelsel wordt voor uw ogen uitgerekend – stap voor stap.

Hoe werkt dat dan? We kiezen als voorbeeld het rekenen in het vijftalstelsel. Als antwoord op de vraag 'Dec., Hex., of Bin.' antwoordt u met het drukken van de (ENTER)-toets. Op de volgende vraag antwoordt u met '5' en toetst nu bijvoorbeeld het getal '230' in. De uitkomst in het vijftalstelsel is '1410'.

Nu ter verklaring van het voorgaande: In het vijftalstelsel hebben we de cijfers '0, 1, 2, 3, en 4'. Uw MSX-computer deelt het door ons opgegeven getal '230' door 5. Uikomst: '46' rest '0'. Aansluitend wordt deze '46' gedeeld: '9' rest '1'. Nu wordt de '9' gedeeld: '1' rest '4' en zo blijft ten slotte een rest '4' over. Zetten we de restwaarden nog eenmaal op een rijtje:

1. 0 2. 1 3. 4

Bij de laatste waarde nemen we niet alleen de rest '4', maar ook het aantal = '1'.

Zo verkrijgen we voor decimaal '230' het getal in het vijftalstelsel n.l. '1410'. Is dat duidelijk? Zo niet, dan staan hieronder nog eens de verschillende verheffingen tot de vijfde macht van rechts naar links:

1) 0 * 5↑0 = 0 2) 1 * 5↑1 = 5 3) 4 * 5↑2 = 100
4) 1 * 5↑3 = 125 –) 0 + 5 + 100 + 125 = 230!

Volledigheidshalve dient nog te worden opgemerkt, dat men ook zonder het programma 'Talstelsels omrekenen' met behulp van MSX-basic altijd decimaal-hexadecimaal en decimaal-binair omrekenen kan en wel met de commando's: 'HEX$(' en 'BIN$('.

```
Invoer (Dec., Hex. of Bin.) ? d255

Decimaal binair/tweetallig hexadecimaal
---------------------------------------
255      00000000 11111111   00FF

Invoer (Dec., Hex of Bin.) ? hffff

Decimaal binair/tweetallig hexadecimaal
---------------------------------------
65535    11111111 11111111   FFFF

Invoer (Dec., Hex. of Bin.) ? b111

Decimaal binair/tweetallig hexadecimaal
---------------------------------------
7        00000000 00000111   0007
```

```
Omrekenen Dec.- 5 talstelsel
============================

Uw opgave (Dec.) ? 230

Aantal: 46    Rest: 0
Aantal: 9     Rest: 1
Aantal: 1     Rest: 4

 1410

Welk talstelsel (2 t/m 9) ?
```

```
10 REM Talstelsels omrekenen
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                         Rainer Lueers
40 COLOR 15,1:SCREEN 0:KEY OFF
50 ON ERROR GOTO 410
60 WIDTH 40
70 REM Keuze uit de talstelsels
80 A$="":INPUT "Invoer(Dec., Hex. of Bin
.)";A$:IF VAL(A$)>65535! THEN GOTO 80 EL
SE IF A$="" THEN GOTO 160 ELSE B$=LEFT$(
A$,1):IF B$="H" OR B$="h" THEN GOSUB 310
 ELSE IF B$="B" OR B$="b" THEN GOSUB 360
 ELSE IF B$="D" OR B$="d" THEN A$=RIGHT$
(A$,LEN(A$)-1)
90 REM Berekenen v.d. Decimale, Binaire
       en hexadecimale getallen voor
       het afbeelden op het scherm
100 PRINT:PRINT "Decimaal binair/tweetal
lig hexadecimaal"
110 PRINT STRING$(39,"-")
120 PRINT A$;TAB(8);" ";
130 B$=STRING$(16-LEN(BIN$(VAL(A$))),"0"
)+BIN$(VAL(A$)):PRINT LEFT$(B$,8);" ";RI
GHT$(B$,8);"  ";
140 PRINT TAB(31);STRING$(4-LEN(HEX$(VAL
(A$))),"0")+HEX$(VAL(A$))
150 PRINT:PRINT:GOTO 80
160 CLS
170 REM Keuze tussen de talstelsels
        2 t/m 9
180 A=0:INPUT "Welk talstelsel (2 t/m9)
";A
190 IF A=0 THEN RUN ELSE IF A<2 OR A>9 T
HEN GOSUB 440:GOTO 160
200 CLS:PRINT "Omrekenen Dec.->";A;"tals
telsel":PRINT STRING$(29,"="):PRINT
210 REM Het omrekenen van de getallen
        in het gewenste talstelsel (2/9)
```

```
220 B=Ø:INPUT "Uw opgave (Dec.) ";B:IF B
>32767 THEN GOSUB 44Ø:GOTO 16Ø ELSE PRIN
T
23Ø C=B\A
24Ø PRINT "Aantal:";C;
25Ø D=B MOD A
26Ø PRINT TAB(14);"Rest:";D
27Ø A$=RIGHT$(STR$(D),1)+A$
28Ø IF C<A THEN A$=STR$(C)+A$:PRINT:PRIN
T A$:A$="":PRINT:PRINT:GOTO 18Ø
29Ø B=C:GOTO 23Ø
3ØØ REM Invoer van het hexadecimale
        getal. Testen en omrekenen
31Ø IF LEN(A$)>5 THEN RETURN
32Ø B$="&h"+MID$(A$,2,LEN(A$)-1)
33Ø IF VAL(B$)<Ø THEN A$=STR$(VAL(B$)+65
536!) ELSE A$=STR$(VAL(B$))
34Ø RETURN
35Ø REM Ivoer van het binaire getal
        testen en omrekenen
36Ø IF LEN(A$)>17 THEN RETURN
37Ø B$="&b"+MID$(A$,2,LEN(A$)-1)
38Ø IF VAL(B$)<Ø THEN A$=STR$(VAL(B$)+65
536!) ELSE A$=STR$(VAL(B$))
39Ø RETURN
4ØØ REM Foutbehandelings routine
41Ø RESUME 42Ø
42Ø RUN
43Ø END
44Ø PRINT TAB(1Ø) CHR$(2ØØ) "verkeerde i
nvoer!" CHR$(2ØØ)
45Ø GOSUB 46Ø:RETURN
46Ø PRINT:PRINT TAB(6) "<Druk een toets
a.u.b.>"
47Ø F$=INKEY$:IF F$="" THEN GOTO 47Ø
48Ø RETURN
```

Disassembler

In de programma's 'Geheugeninhoud 1' tot en met 'Geheugeninhoud 5' alsmede in het programma 'Variabelenreferentielijst' hebben we ons al een klein beetje met het 'binnenwerk' van onze MSX-computer beziggehouden. Onder 'binnenwerk' verstaan we: Hoe is het eigenlijk mogelijk dat een onmenselijk ding als een computer tot denken in staat is (bedenk daarbij het volgende: Een computer is altijd net zo slim als de mensen die hem geprogrammeerd hebben.)

In de programma's 'Geheugeninhoud 1' tot en met 'Geheugeninhoud 5' zijn we echter nog niet ver genoeg in het 'binnenwerk' van onze MSX-computer doorgedrongen. We hebben toen namelijk alleen maar gekeken hoe onze BASIC-programma's regelnummer voor regelnummer, commando voor commando, per geheugenplaats werden opgeslagen. Maar we hebben niets geleerd over de verdere verwerking van deze commando's. Hierbij werd echter al aangetoond, dat de MSX-computer, zoals iedere andere computer, gebaseerd is op het basisprincipe: aan of uit, ofwel in binaire vorm: nul of één. Door het herkennen van een aantal opeenvolgende nullen en/of enen constateert de computer dan een teken, vaak ook een commando.

Onze BASIC noemt men een hogere programmeertaal: deze bestaat uit een groot aantal verstandelijke begrippen ofwel commando's, zodat we tijdens het gebruik van deze commando's ons veel beter kunnen voorstellen wat de computer doen kan, dan wanneer we alleen nullen en enen per geheugenplaats zouden inbrengen.

De in dit programma onderzochte assemblercode is een stap verder op de weg van BASIC naar machinetaal. We hebben hier een groot aantal commando's ter beschikking (bij de Z80A processor in de MSX-computer zijn dat er zelfs meer dan 600), die echter bij lange na niet zo'n effectief resultaat geven als de BASIC-commando's. Per geheugenpositie bezien hebben deze commando's deels 1 byte, deels 2 bytes, nodig. Vaak ook echter 3 bytes (1 byte = 1 geheugenplaats). Het programma 'disassembler' kent alle Z80A-assemblercodes (die men mnemonics noemt) en onderzoekt de door ons opgegeven geheugenplaatsen of deze daar ook daadwerkelijk voorkomen. Daarbij zoekt het programma naar het voorkomen van bepaalde getalreeksen (op dezelfde manier als bij het programma 'Geheugeninhoud 4', waar we naar de reeks 'lengte' gezocht hebben). Worden de getalreeksen ergens aangetroffen, dan wordt de betreffende assemblercode op het beeldscherm of de printer weergegeven.

Omdat we niet veronderstellen dat u dagelijks zeer veel machinetaalprogramma's schrijft, hebben we een leerzaam effect in ons 'disassembler-programma' ingebouwd. In regel 3560 staat een zgn. DATA-commando, gevolgd door enkele hexadecimale getallen. Deze getallen vormen een klein machinetaalprogramma, waarbij twee getallen worden opgeteld en op een bepaalde geheugenlocatie worden opgeslagen. U kunt het omzetten van deze getallen met een assemblerprogramma volgen. Na het starten van het programma antwoordt u op alle vragen zonder verdere invoer slechts met (ENTER). Alleen de vraag 'printer of beeldscherm' beantwoordt u met 'B' voor beeldscherm, indien u geen schriftelijke uitdraai nodig heeft.

U kunt nu ook andere getallen (in hexadecimale notatie) in de dataregel invoeren, of u vindt een langer machinetaalprogramma voor de Z80A-processor in een computertijdschrift . . . U kunt ook meerdere DATA-regels met getallen toevoegen . . . Veel plezier ermee!

```
10 REM Disassembler
20 REM MSX Verzamelde programma's
30 REM copyright 1984 DATA BECKER &
                          Rainer Lueers
40 REM Geheugenplaatsen vanaf 55000
       voor machineroutines vrij houden
50 CLEAR 2000,54999!:GOSUB 3470
60 DIM I$(255),BL(255):N=0:M=0:SY=1
70 KEY OFF:CLS:GOTO 1290
80 SCREEN 0:WIDTH 40:COLOR 15,1:C=0:PRIN
T TAB(3) "* * * Z-80 Disassembler * * *"
90 REM Wensen van de gebruiker bezien
100 PRINT:INPUT "titel ";F$
110 IF LEN(F$)>80 THEN GOTO 100
120 PRINT
130 REM Startadres van de machinetaal-
        routine in de DATA-regels
140 A=0:INPUT "Startadres (dec.) ";A:IF
A<0 OR A>65535! THEN GOTO 120 ELSE IF A=
0 THEN A=55000!
150 INPUT "Eindadres (dec.)    ";B
160 IF B<E OR B>65535! OR B<0 THEN GOTO
150 ELSE IF B=0 THEN B=ZA
170 PRINT: INPUT "Data-regels aanwezig (
J/ ) ";V$
180 IF LEFT$(V$,1)="J" OR LEFT$(V$,1)="j
" THEN V$="J"
190 IF V$="J" THEN GOSUB 2150
200 INPUT "Printer op beeldscherm (P/B)
";B$
210 REM Benoemen van de variabele b$
        die voor het aansturen van het
        print of lprint commando
        bepalend is
220 IF LEFT$(B$,1)="P" OR LEFT$(B$,1)="p
" THEN B$="P" ELSE IF LEFT$(B$,1)="b" OR
 LEFT$(B$,1)="B" THEN B$="B"
230 IF B$="P" THEN LPRINT ELSE PRINT
```

```
240 REM Opschrift = titel die in
      in het centrum v.d. regel
  wordt afgebeeld
250 IF B$="P" THEN LPRINT TAB((80-LEN(F$
))/2-5) F$ ELSE PRINT TAB((40-LEN(F$))/2
-5) F$
260 IF B$="P" THEN LPRINT ELSE PRINT
270 REM Het begin van de eigenlijke
        disassembler. Successievelijk
        inlezen van de aangegeven
        geheugenplaatsen en het
        omzetten van de data in machine-
        taalcommando's = Mnemonics
280 Q=A
290 H1=INT(A/4096):H2=INT((A-H1*4096)/25
6)
300 L1=INT((A-H1*4096-H2*256)/16)
310 L2=A-H1*4096-H2*256-L1*16
320 IF M<>0 THEN GOTO 340
330 GOTO 390
340 IF A>=A(N) AND A<=B(N) THEN GOTO 360
350 GOTO 390
360 D=PEEK(A):GOSUB 3390:F$="D":FL$="A":
GH$="T":GL$="A":T$=I$(D):I$(D)=""
370 IF A=B(N) THEN N=N+1:M=M-1:GOTO 400
380 GOTO 400
390 D=PEEK(A):GOSUB 660
400 V=D:GOSUB 2750
410 DH$=H$:DL$=L$
420 IF H1>=10 THEN GOTO 570
430 H1$=STR$(H1):H1$=MID$(H1$,2,1)
440 IF H2>=10 THEN GOTO 580
450 H2$=STR$(H2):H2$=MID$(H2$,2,1)
460 IF L1>=10 THEN GOTO 590
470 L1$=STR$(L1):L1$=MID$(L1$,2,1)
480 IF L2>=10 THEN GOTO 600
490 L2$=STR$(L2):L2$=MID$(L2$,2,1)
500 REM Uitvoer van het onderzochte
        geheugengebied
```

```
510 IF B$="P" THEN QT=QT+1:IF QT>105 THE
N QT=Ø:FOR QI=1 TO 9:LPRINT:NEXT QI:GOSU
B 341Ø:SY=SY+1:LPRINT TAB(3Ø) "Pagin";SY
:LPRINT:GOTO 51Ø ELSE LPRINT Q;LEN(DH$+D
L$+EH$+EL$+FH$+FL$+GH$+GL$)/2;"bytes";TA
B(18);
52Ø IF B$="P" THEN LPRINT USING "    !!!!
";" ",H1$,H2$,L1$,L2$;:LPRINT "
  ";:LPRINT USING "!!!!!!!!!";" ",DH$,DL$
,EH$,EL$,FH$,FL$,GH$,GL$;:LPRINT TAB(5Ø)
;I$(D)
53Ø IF B$="B" THEN PRINT USING "!!!!";"
",H1$,H2$,L1$,L2$;:PRINT "  ";:PRINT USI
NG "!!!!!!!!!";" ",DH$,DL$,EH$,EL$,FH$,FL
$,GH$,GL$;:PRINT TAB(15);I$(D)
54Ø A=A+1:C=C+1:IF A>=B+1 THEN GOTO 61Ø
55Ø IF GH$="T" THEN I$(D)=T$
56Ø GOTO 28Ø
57Ø X=H1-1Ø+65:H1$=CHR$(X):GOTO 44Ø
58Ø X=H2-1Ø+65:H2$=CHR$(X):GOTO 46Ø
59Ø X=L1-1Ø+65:L1$=CHR$(X):GOTO 48Ø
6ØØ X=L2-1Ø+65:L2$=CHR$(X):GOTO 51Ø
61Ø PRINT:PRINT "      * * Weergave gere
ed  * *"
62Ø REM Vragen naar eventuele herstart
63Ø PRINT:F$="":INPUT "Nog een keer (J/
) ";F$:IFLEFT$(F$,1)="j" OR LEFT$(F$,1)=
"J" THEN F$="J"
64Ø IF F$="J" THEN RUN
65Ø END
66Ø IF D<64 OR D>127 THEN GOTO 72Ø
67Ø IF D=118 THEN RETURN
68Ø DH=INT(D/16):DL=D-DH*16
69Ø G=DL AND 7:F=((D AND 56)/8)
7ØØ I$(D)="ld          "+J$(F)+","
71Ø I$(D)=I$(D)+J$(G):GOSUB 339Ø:RETURN
72Ø IF D<128 OR D>191 THEN GOTO 84Ø
73Ø DH=INT(D/16):DL=D-(DH*16)
74Ø G=DL AND 7:F=((D AND 12Ø)/8)
```

```
750 IF F=0 THEN I$(D)="ADD         A,"
760 IF F=1 THEN I$(D)="ADC         A,"
770 IF F=2 THEN I$(D)="SUB         "
780 IF F=3 THEN I$(D)="SBC         A,"
790 IF F=4 THEN I$(D)="AND         "
800 IF F=5 THEN I$(D)="XOR         "
810 IF F=6 THEN I$(D)="OR          "
820 IF F=7 THEN I$(D)="CP          "
830 I$(D)=I$(D)+J$(G):GOSUB 3390:RETURN
840 IF BL(D)=1 THEN GOTO 890
850 IF BL(D)=2 THEN GOTO 900
860 IF BL(D)=3 THEN GOTO 910
870 IF BL(D)=4 THEN GOTO 1780
880 IF D=221 OR D=253 THEN GOTO 2230
890 GOSUB 3390:RETURN
900 A=A+1:Z=PEEK(A):GOTO 920
910 A=A+1:Z=PEEK(A):A=A+1:Z1=PEEK(A)
920 V=Z:GOSUB 2750:EH$=H$:EL$=L$
930 IF BL(D)=3 THEN GOTO 950
940 GOSUB 3400:GOTO 960
950 V=Z1:GOSUB 2750:FH$=H$:FL$=L$
960 P1$=FH$+FL$+EH$+EL$:P2$=EH$+EL$:GH$=
"":GL$=""
970 F=D AND 7:G=D AND 56:G=G/8
980 IF F=6 AND (D AND 192)=0 THEN GOTO 1
240
990 IF F=2 AND (D AND 192)=192 THEN GOTO
 1270
1000 IF F=4 THEN GOTO 1280
1010 IF F=0 AND (G<>2) THEN GOTO 1250
1020 I$(33)="LD          HL,"+P1$
1030 I$(34)="LD          ("+P1$+"),HL"
1040 I$(50)="LD          ("+P1$+"),A"
1050 I$(205)="CALL        "+P1$
1060 I$(195)="JP          "+P1$
1070 I$(58)="LD          A,("+P1$+")"
1080 I$(254)="CP          "+P2$
1090 I$(42)="LD          HL,("+P1$+")"
1100 I$(49)="LD          SP,"+P1$
```

```
111Ø I$(17)="LD          DE,"+P1$
112Ø I$(16)="DJNZ         "+P2$
113Ø I$(1)="LD          BC,"+P1$
114Ø I$(198)="ADD          A,"+P2$
115Ø I$(2Ø6)="ADC          A,"+P2$
116Ø I$(211)="OUT          "+P2$+",A"
117Ø I$(214)="SUB          "+P2$
118Ø I$(219)="IN           A,"+P2$
119Ø I$(222)="SBC          A,"+P2$
12ØØ I$(23Ø)="AND          "+P2$
121Ø I$(238)="XOR          "+P2$
122Ø I$(246)="OR           "+P2$
123Ø RETURN
124Ø I$(D)="LD           "+J$(G)+","+P2$:RE
TURN
125Ø IF G=3 THEN I$(D)="JR           "+P2$:
RETURN
126Ø G=G-4:I$(D)="JR           "+S$(G)+","+
P2$:RETURN
127Ø I$(D)="JP           "+S$(G)+","+P1$:RE
TURN
128Ø I$(D)="CALL         "+S$(G)+","+P1$:RE
TURN
129Ø FOR S=Ø TO 63:BL(S)=1:NEXT S
13ØØ FOR S=192 TO 255:BL(S)=1:NEXT S
131Ø BL(118)=1:BL(6)=2:BL(14)=2:BL(16)=2
:BL(22)=2:BL(24)=2:BL(3Ø)=2:BL(32)=2:BL(
38)=2:BL(4Ø)=2:BL(46)=2:BL(48)=2:BL(54)=
2:BL(56)=2
132Ø BL(62)=2:BL(198)=2:BL(2Ø6)=2:BL(211
)=2:BL(214)=2:BL(219)=2:BL(222)=2:BL(23Ø
)=2:BL(238)=2:BL(246)=2:BL(254)=2
133Ø BL(1)=3:BL(17)=3:BL(33)=3:BL(34)=3:
BL(42)=3:BL(49)=3:BL(5Ø)=3:BL(58)=3:BL(1
94)=3:BL(195)=3:BL(196)=3:BL(2Ø2)=3:BL(2
Ø4)=3:BL(2Ø5)=3:BL(21Ø)=3:BL(212)=3:BL(2
18)=3:BL(22Ø)=3
```

```
1340 BL(226)=3:BL(228)=3:BL(234)=3:BL(23
6)=3:BL(242)=3:BL(244)=3:BL(250)=3:BL(25
2)=3:BL(203)=3:BL(237)=3:BL(221)=3:BL(25
3)=3
1350 J$(0)="B":J$(1)="C":J$(2)="D":J$(3)
="E":J$(4)="H":J$(5)="L"
1360 J$(6)="(HL)":J$(7)="A"
1370 S$(0)="NZ":S$(1)="Z":S$(2)="NC":S$(
3)="C":S$(4)="PO":S$(5)="PE":S$(6)="P":S
$(7)="M"
1380 P$(0)="SBC         HL,":P$(1)="ADC
     HL,"
1390 P$(2)="SBC         HL,":P$(3)="ADC
     HL,"
1400 P$(4)="SBC         HL,":P$(5)="ADC
     HL,"
1410 P$(7)="ADC         HL,":N$(0)="BC":N$
(1)="BC":N$(2)="DE"
1420 N$(3)="DE":N$(4)="HL":N$(5)="HL":N$
(7)="SP"
1430 L$(0)="N":L$(1)="I":M$(0)="0":M$(2)
="1":M$(3)="2"
1440 O$(0)="I,A":O$(1)="R,A":O$(2)="A,I"
:O$(3)="A,R"
1450 Q$(0)="LD":Q$(1)="CP":Q$(2)="IN":Q$
(3)="OUT"
1460 I$(O)="NOP":I$(2)="LD          (BC),A
":I$(3)="INC         BC"
1470 I$(7)="RLC         A":I$(8)="EX
  AF,AF'"
1480 I$(9)="ADD         HL,BC":I$(10)="LD
        A,(BC)"
1490 I$(11)="DEC         BC":I$(15)="RCC
     A"
1500 I$(18)="LD          (DE),A":I$(19)="I
NC          DE"
1510 I$(23)="RLA":I$(25)="ADD         HL,D
E":I$(26)="LD          A,(DE)"
1520 I$(27)="DEC         DE":I$(31)="RRA":
I$(35)="INC         HL"
```

```
1530 I$(39)="DAA":I$(41)="ADD        HL,H
L":I$(43)="DEC        HL"
1540 I$(47)="CPL":I$(51)="INC        SP":
I$(55)="SCF"
1550 I$(57)="ADD        HL,SP":I$(59)="DE
C        SP":I$(63)="CCF"
1560 I$(43)="DEC        HL"
1570 I$(192)="RET        NZ":I$(197)="PUS
H       BC"
1580 I$(199)="RST        0":I$(200)="RET
       Z":I$(207)="RST        8"
1590 I$(208)="RET        NC":I$(209)="POP
       DE"
1600 I$(213)="PUSH       DE":I$(215)="RST
       10H"
1610 I$(216)="RET        C":I$(223)="RST
      18H"
1620 I$(224)="RET        PO":I$(227)="EX
       (SP),HL"
1630 I$(231)="RST        20H":I$(232)="RE
R        PE"
1640 I$(233)="JP         (HL)":I$(235)="E
X         DE,HL"
1650 I$(239)="RST        28H":I$(240)="RE
T        P"
1660 I$(201)="RET"
1670 I$(241)="POP        AF":I$(243)="DI"
:I$(245)="PUSH       AF"
1680 I$(247)="RST        30H":I$(248)="RE
T        M":I$(251)="EI"
1690 I$(249)="LD         SP,HL":I$(255)="
RST        38H"
1700 I$(197)="PUSH       BC"
1710 I$(193)="POP        BC":I$(229)="PUS
H       HL"
1720 I$(225)="POP        HL"
1730 I$(217)="EXX"
1740 I$(118)="HALT"
1750 L=0:FOR D=5 TO 45 STEP 8:I$(D)="DEC
       "+J$(L):L=L+1:NEXT D
```

```
1760 L=0:FOR D=4 TO 44 STEP 8:I$(D)="INC
        "+J$(L):L=L+1:NEXT D
1770 I$(60)="INC        A":I$(61)="DEC
     A":GOTO 80
1780 IF D=203 THEN GOTO 1800
1790 IF D=237 THEN GOTO 2960
1800 A=A+1:Z=PEEK(A)
1810 EH=INT(Z/16):EL=Z-EH*16:G=EL AND 7
1820 F=INT((EL AND 8)/8):F=F+(EH*2):GOSU
B 1840
1830 GOTO 1990
1840 IF F=0 THEN I$(D)="RLC        ":RETU
RN
1850 IF F=1 THEN I$(D)="RRC        ":RETU
RN
1860 IF F=2 THEN I$(D)="RL         ":RETU
RN
1870 IF F=3 THEN I$(D)="RR         ":RETU
RN
1880 IF F=4 THEN I$(D)="SLA        ":RETU
RN
1890 IF F=5 THEN I$(D)="SRA        ":RETU
RN
1900 IF F=7 THEN I$(D)="SRL        ":RETU
RN
1910 IF F>=8 AND F<=15 THEN GOTO 1950
1920 IF F>=16 AND F<=23 THEN GOTO 1970
1930 F=F AND 7:F$=CHR$(F+48)
1940 I$(D)="SET       "+F$+",":RETURN
1950 F=F AND 7:F$=CHR$(F+48)
1960 I$(D)="BIT       "+F$+",":RETURN
1970 F=F AND 7:F$=CHR$(F+48)
1980 I$(D)="RES       "+F$+",":RETURN
1990 IF EH>=10 THEN GOTO 2040
2000 EH$=STR$(EH):EH$=MID$(EH$,2,1)
2010 IF EL>=10 THEN GOTO 2050
2020 EL$=STR$(EL):EL$=MID$(EL$,2,1)
2030 GOSUB 3400:I$(D)=I$(D)+J$(G):RETURN
2040 X=EH+55:EH$=CHR$(X):GOTO 2010
2050 X=EL+55:EL$=CHR$(X):GOTO 2030
```

```
2060 IF D AND 7-5 THEN GOTO 2090
2070 IF D AND 7-4 THEN GOTO 2090
2080 GOTO 890
2090 L=D AND 56
2100 G=L/8
2110 IF D AND 7=5 THEN I$(D)="DEC
"+J$(G):GOTO 2130
2120 I$(D)="INC        "+J$(G)
2130 GOSUB 3390:RETURN
2140 REM Routine t.e. disassembleren
         van bepaalde geheugenbereiken
         te voorkomen
2150 N=1
2160 INPUT "Startad.=";A(N)
2170 INPUT "Eindadr.=";B(N)
2180 IF A(N)<A OR B(N)>B THEN GOTO 2160
2190 INPUT "Nog een deel (J/ ) ";V$
2200 IF LEFT$(V$,1)="j" OR LEFT$(V$,1)="
J" THEN V$="J"
2210 IF V$="J" THEN N=N+1:GOTO 2160
2220 M=N:N=1:RETURN
2230 IF D=221 THEN V$="IX":GOTO 2250
2240 V$="IY"
2250 A=A+1:Z=PEEK(A)
2260 IF Z=203 THEN GOTO 2670
2270 IF Z>=70 AND Z<=190 THEN GOTO 2820
2280 IF Z=33 OR Z=34 OR Z=42 THEN GOTO 2
440
2290 IF Z=52 OR Z=53 THEN GOTO 2540
2300 IF Z=54 THEN GOTO 2600
2310 V=Z:GOSUB 2750:EH$=H$:EL$=L$:GOSUB
3400
2320 IF Z=9 THEN I$(D)="ADD          "+V$+
",BC"
2330 IF Z=25 THEN I$(D)="ADD         "+V$+
",DE"
2340 IF Z=35 THEN I$(D)="INC         "+V$
2350 IF Z=41 THEN I$(D)="ADD         "+V$+
","+V$
2360 IF Z=43 THEN I$(D)="DEC         "+V$
2370 IF Z=57 THEN I$(D)="ADD         "+V$+
",SP"
```

```
2380 IF Z=225 THEN I$(D)="POP         "+V$
2390 IF Z=227 THEN I$(D)="EX          (SP)
,"+V$
2400 IF Z=229 THEN I$(D)="PUSH        "+V$
2410 IF Z=233 THEN I$(D)="JP          ("+V
$+")"
2420 IF Z=249 THEN I$(D)="LD          SP,"
+V$
2430 RETURN
2440 V=Z:GOSUB 2750
2450 EH$=H$:EL$=L$
2460 A=A+1:Z1=PEEK(A):V=Z1:GOSUB 2750
2470 FH$=H$:FL$=L$:A=A+1
2480 Z1=PEEK(A):V=Z1:GOSUB 2750
2490 GH$=H$:GL$=L$
2500 IF Z=33 THEN I$(D)="LD          "+V$+
","+GH$+GL$+FH$+FL$
2510 IF Z=34 THEN I$(D)="LD          ("+GH
$+GL$+FH$+FL$+"),"+V$
2520 IF Z=42 THEN I$(D)="LD          "+V$+
",("+GH$+GL$+FH$+FL$+")"
2530 RETURN
2540 V=Z:GOSUB 2750:EH$=H$:EL$=L$
2550 A=A+1:Z1=PEEK(A):V=Z1:GOSUB 2750
2560 FH$=H$:FL$=L$:GH$="":GL$=""
2570 IF Z=52 THEN I$(D)="INC          ("+V
$+"+"+FH$+FL$+")"
2580 IF Z=53 THEN I$(D)="DEC          ("+V
$+"+"+FH$+FL$+")"
2590 RETURN
2600 V=Z:GOSUB 2750:EH$=H$:EL$=L$
2610 A=A+1:Z=PEEK(A):V=Z1:GOSUB 2750
2620 FH$=H$:FL$=L$:A=A+1
2630 Z1=PEEK(A):V=Z1:GOSUB 2750
2640 GH$=H$:GL$=L$
2650 I$(D)="LD          ("+V$+"+"+FH$+FL$+
"),"+GH$+GL$
2660 RETURN
2670 A=A+1:EH$=CHR$(67):EL$=CHR$(66)
2680 V=PEEK(A):GOSUB 2750
2690 FH$=H$:FL$=L$
```

```
2700 A=A+1:Z2=PEEK(A):O=Z2 AND 248:O=O/8
2710 Q$="("+V$+"+"+FH$+FL$+")"
2720 F=O:GOSUB 1840
2730 I$(D)=I$(D)+Q$
2740 V=Z2:GOSUB 2750:GH$=H$:GL$=L$:RETUR
N
2750 H=INT(V/16):L=V-(H*16)
2760 IF H>=10 THEN GOTO 2800
2770 H$=STR$(H):H$=MID$(H$,2,1)
2780 IF L>=10 THEN GOTO 2810
2790 L$=STR$(L):L$=MID$(L$,2,1):RETURN
2800 X1=H+55:H$=CHR$(X1):GOTO 2780
2810 X1=L+55:L$=CHR$(X1):RETURN
2820 V=Z:GOSUB 2750:EH$=H$:EL$=L$:GH$=""
:GL$=""
2830 A=A+1:Z1=PEEK(A)
2840 V=Z1:GOSUB 2750:FH$=H$:FL$=L$
2850 IF Z=126 THEN I$(D)="LD         A,("
+V$+"+"+FH$+FL$+")":RETURN
2860 P=Z AND 240
2870 IF P=112 THEN GOTO 2910
2880 IF P>=128 THEN GOTO 2930
2890 P=Z AND 56:P=P/8:GOSUB 3300
2900 I$(D)="LD          "+G$+",("+V$+"+"+F
H$+FL$+")":RETURN
2910 P=Z AND 7:GOSUB 3300
2920 I$(D)="LD          ("+V$+"+"+FH$+FL$+
"),"+G$:RETURN
2930 P=Z AND 56:P=P/8:GOSUB 3300
2940 I$(D)=I$(D)+V$+"+"+FH$+FL$+")"
2950 RETURN
2960 A=A+1:Z=PEEK(A):V=Z:GOSUB 2750:EH$=
H$:EL$=L$
2970 IF Z=67 OR Z=75 OR Z=83 OR Z=91 OR
Z=115 OR Z=123 THEN GOTO 3210
2980 GOSUB 3400:F=Z AND 248:G=Z AND 7
2990 IF F=160 THEN GOTO 3170
3000 IF F=168 THEN GOTO 3180
3010 IF F=176 THEN GOTO 3190
3020 IF F=184 THEN GOTO 3200
3030 F=Z AND 56:F=F/8:G=Z AND 7
```

```
3040 IF F=6 THEN I$(D)="SBC        HL,SP"
:RETURN
3050 IF G=0 THEN I$(D)="IN         "+J$(F
)+",(C)":RETURN
3060 IF G=1 THEN I$(D)="OUT        (C),"+
J$(F):RETURN
3070 IF G=2 THEN I$(D)=P$(F)+N$(F):RETUR
N
3080 IF G=4 THEN I$9#)="NEG":RETURN
3090 IF G=5 THEN I$(D)="RET"+L$(F):RETUR
N
3100 IF G=6 THEN I$(D)="IM "+M$(F):RETUR
N
3110 IF G<>7 THEN I$(D)="* * *":RETURN
3120 IF F>3 THEN GOTO 3160
3130 IF F=4 THEN I$(D)="RRD":RETURN
3140 IF F=5 THEN I$(D)="RLD":RETURN
3150 GOTO 3110
3160 I$(D)="LD         "=O$(F):RETURN
3170 I$(D)=Q$(G)+"I":RETURN
3180 I$(D)=Q$(G)+"D":RETURN
3190 I$(D)=Q$(G)+"IR":RETURN
3200 I$(D)=Q$(G)+"DR":RETURN
3210 A=A+1:Z1=PEEK(A):V=Z1:GOSUB 2750:FH
$=H$:FL$=L$
3220 A=A+1:Z1=PEEK(A):V=Z1:GOSUB 2750:GH
$=H$:GL$=L$
3230 GG$=GH$+GL$+FH$+FL$
3240 IF Z=67 THEN I$(D)="LD        ("+GG
$+"),BC":RETURN
3250 IF Z=75 THEN I$(D)="LD        BC,("
+GG$+")";RETURN
3260 IF Z=83 THEN I$(D)="LD        ("+GG
$+"),DE":RETURN
3270 IF Z=91 THEN I$(D)="LD        DE,("
+GG$+")":RETURN
3280 IF Z=115 THEN I$(D)="LD        ("+G
G$+"),SP":RETURN
3290 IF Z=123 THEN I$(D)="LD        SP,(
"+GG$+")":RETURN
```

```
3300 IF P=0 THEN G$="B":I$(D)="ADD
  A,("
3310 IF P=1 THEN G$="C":I$(D)="ADC
A,("
3320 IF P=2 THEN G$="D":I$(D)="SUB      (
"
3330 IF P=3 THEN G$="E":I$(D)="SBC      A
,("
3340 IF P=4 THEN G$="H":I$(D)="AND      (
"
3350 IF P=5 THEN G$="L":I$(D)="XOR      (
"
3360 IF P=6 THEN G$="***":I$(D)="OR
 ("
3370 IF P=7 THEN G$="A":I$(D)=CP        (
"
3380 RETURN
3390 EH$="":EL$="":FH$="":FL$="":GH$="":
GL$="":RETURN
3400 FH$="":FL$="":GH$="":GL$="":RETURN
3410 IF B$="P" THEN LPRINT ELSE PRINT
3420 QT=0:SY=1
3430 IF B$="P" THEN LPRINT ELSE PRINT
3440 RETURN
3450 REM Einde van de programma-
         bibliotheek voor het
         reproduceren van de
         assemblercodes
3460 REM Het testen van het aantal
         te bewerken bytes (hexadec.
         invoer) in de DATA-regels
3470 ON ERROR GOTO 3560:FOR ZA=55000! TO
 65535!:READ A$:NEXT ZA
3480 REM Het feitelijke inlezen van de
         bytes in de DATA-regels in het
         geheugen vanaf locatie 43776
3490 RESTORE:FOR M=55000! TO ZA-1
3500 READ A$
3510 A=VAL("&h"+A$)
3520 POKE M,A
3530 NEXT M:POKE M,&HC9
```

```
3540 RETURN
3550 DATA 3e,04,06,07,80,32,00,7d
3560 IF ERR=4 THEN RESUME 3490 ELSE STOP
```

```
    + + + Z-80 Disassembler + + +

Titel? Eenvoudige optelling in MC

Startadres (in Dec.) ?
Eindadres (in Dec.)  ?

Datavelden voorhanden (J/ ) ?

Printer of Beeldscherm (P/B) ? b

    Eenvoudige optelling in MC

D6D8   3E04    LD    A,04
D6DA   0607    LD    B,07
D6DC   80      ADD   A,B
D6DD   32007D  LD    (7D00),A
D6E0   C9      RET

+ + uitvoer klaar + +

3560 DATA 3e,04,06,07,80,32,00,7d
Ok
```

Appendix

De besturingskarakters van de MSX-computer (zie 'geheugeninhoud 1 t/m geheugeninhoud 5')

32 tot en met 126: normale ASCII-karakterset.

129 END	130 FOR	131 NEXT	132 DATA
133 INPUT	134 DIM	135 READ	136 LET
137 GOTO	138 RUN	139 IF	140 RESTORE
141 GOSUB	142 RETURN	143 REM	144 STOP
145 PRINT	146 CLEAR	147 LIST	148 NEW
149 ON	150 WAIT	151 DEF	152 POKE
153 CONT	154 CSAVE	155 CLOAD	156 OUT

157 LPRINT	158 LLIST	159 CLS	160 WIDTH
161 LSE	162 TRON	163 TROF	164 SWAP
165 ERASE	166 ERROR	167 RESUME	168 DELETE
169 AUTO	170 RENUM	171 DEFSTR	172 DEFINT
173 DEFSNG	174 DEFDBL	175 LINE	176 OPEN
177, FIELD	178 GET	179 PUT	180 CLOSE
181 LOAD	182 MERGE	183 FILES	184 LSET
185 RSET	186 SAVE	187 LFILES	188 CIRCLE
189 COLOR	190 DRAW	191 PAINT	192 BEEP
193 PLAY	194 PSET	195 PRESET	196 SOUND
197 SCREEN	198 VPOKE 199	SPRITE	200 VDP
201 BASE	202 CALL	203 TIME	204 KEY
205 MAX	206 MOTOR	207 BLOAD	208 BSAVE
209 DSKO$	210 SET	211 NAME	212 KILL
213 IPL	214 COPY	215 CMD	216 LOCATE
217 TO	218 THEN	219 TAB(	220 STEP
221 USR	222 FN	223 SPC(	224 NOT
225 ERL	226 ERR 227	STRING$	228 USING
229 INSTR	230 '(geen ASCII,	maar commando 'REM')	
231 VARPTR	232 CSRLIN	233 ATTR$	234 DSKI$
235 OFF	236 INKEY$	237 POINT	238 > (rek.)
239 = (rek.)	240 < (rek.)	241 + (rek.)	242 - (rek.)
243 * (rek.)	244 / (rek.)	245 (rek.)	246 AND
247 OR	248 XOR	249 EQV	250 IMP

251 MOD252 tot 254: geen zichtbare besturingskarakters.

Met het teken '255' ervoor ontstaan de volgende besturingskarakters:

1 LEFT$	2 RIGHT$	3 MID$	4 SGN
5 INT	6 ABS	7 SQR	8 RND
9 SIN	10 LOG	11 EXP	12 COS
13 TAN	14 ATN	15 FRE	16 INP
17 POS	18 LEN	19 STR$	20 VAL
21 ASC	22 CHR$	23 PEEK	24 VPEEK
25 SPACE$	26 OCT$	27 HEX$	28 LPOS
29 BIN$	30 CINT	31 CSNG	32 CDBL
33 FIX	34 STICK	35 STRIG	36 PDL
37 PAD	38 DSKF	39 FPOS	40 CVI
41 CVS	42 CVD	43 EOF	44 LOC
45 LOF	46 MKI$	47 MKS$	48 MKD$

N.B. de afkorting (rek.) staat voor rekenkundig.

Deze programmaverzameling is veelomvattend: van een disassembler tot een sporttabel voor de standen in de eerste en eredivisie van de voetbalcompetitie. Ook zijn in dit boek spelletjes en complete utilities te vinden. De listings die in dit boek zijn opgenomen, zijn alle voorzien van een duidelijke uitleg over de handige programmeertrucs die erin verwerkt zijn.
Alle in dit boek opgenomen programma's zijn geschikt voor MSX-computers alsmede voor de SPECTRAVIDEO 318 en 328.

Uit de inhoud:
hexdump ✱ grafiek editor ✱ geluid ✱ Umlauten op het scherm ✱ crossreference voor variabelen ✱ kalender ✱ disassembler ✱ gegevensverwerking/langspeelplatenbestand ✱ hollow, het kersenspel ✱ balkdiagrammen - sporttabel ere- en eerste divisie voetbalcompetitie.

ISBN 90 229 3350 4

MSX Bibliotheek 1

DATA BECKER NEDERLANDS ✱